7-8

1978

P. 1897/78
POLITECHNIKA ŚLĄSKA BIBLIOTEKA GŁÓWNA Gliwice

informatyka

W NUMERZE:

strona

Komputery JS EMC w ZETO Katowice — doświadczenia eksploatacyjne. Część 1. Sprzęt i oprogramowanie podstawowe
Bolesław Gliksman 1

Transliteracja alfabetu rosyjskiego na łaciński w systemach informatycznych
Stefan J. Niementowski 5

PROGRES PR-5 — system planowania i kontroli realizacji prac naukowo-badawczych
Michał Ziębiński 8

Prawa twórców programów
Jan Waluszewski 11

Język SPECOL narzędziem użytkownika
Andrzej Kukuła 15

Problemy projektowania zbioru branżowego
Bolesław Śliz 17

SYWIR — system wyszukiwania informacji i redagowania biuletynów
Zbigniew Odrowąż-Sypniewski, Michał Ostrowski 19

Jednolity system informatyczny w spółdzielczości mieszkaniowej
Krystyn Bernatowicz 22

Metoda HIPO (2) 24

SZTUCZNA INTELIGENCJA

Problemy i decyzje
Marek Hołyński 27

ZE ZJEDNOCZENIA INFORMATYKI

SEIK — system ewidencji informacji
oprac. Andrzej Klimek 31

Pakiet programowy JAZ 75 35

Warunki sprzedaży produktów OBRI 36

Współpraca z jednostkami resortu finansów
Jerzy Bednarz, Zenon Mniszak 37

Białostockie Dni Informatyki 39

Jeszcze o nartach 40

Przed pierwszym setem 40

NAUCZANIE I SZKOLENIE

Komputer w szkole
Andrzej Klimek 41

Z KRAJU

Pokaz sprzętu pomocniczego
Władysław Klepacz 42

Prawie debiut, a jakże udany
Krystyn Bernatowicz 43

Informatyka w ochronie środowiska 44

Popularyzacja opracowań 44

Kompilator PASCALA w konwersacyjnym systemie CMS
Zofia Dzieniszewska, Grażyna Flejta 45

USPRAWNIENIA, NOWE KONCEPCJE, POMYSŁY

Usprawnienie programu sprowadzającego inne programy do pamięci operacyjnej w systemie ODRA 1300
Piotr Gizbert-Studnicki, Jerzy Ptak 46

O wykorzystaniu symbolu jednostki miary
Piotr Omieciński 47

ZE ŚWIATA

Przemysł informatyczny NRD na Wiosennych Targach Lipskich 1978 48

Głosowe urządzenie wejściowe firmy NEC
oprac. Z. Naotyński 50

System VIEWDATA
J. Zembrzuski 50

Wydruk na papierze to jeszcze nie wynik epd! 51

OŚRODKI INFORMATYKI PREZENTUJĄ

System GEORGE 2 — doświadczenia eksploatacyjne
Piotr Brzezicki, Roman Stygar 52

NASZE RECENZJE

Problemy układania harmonogramów
Maciej M. Sysło 54

Jak ZSAP wspomagają prace projektowe
Stanisława Bońkowicz-Sittauer 55

PROBLEMATYKA BAZY DANYCH

Bazy danych i banki danych
Zygmunt Ryznar 57

WYDAWNICTWA CZASOPISM TECHNICZNYCH NOT
Warszawa
Czackiego 3/5

KOLEGIUM REDAKCYJNE

Redaktor Naczelny prof. dr hab. Leon ŁUKASZEWICZ
mgr Krystyn BERNATOWICZ, prof. dr hab. inż. Konrad FIAŁKOWSKI (zast. redaktora naczelnego), mgr Janusz GWIAZDA, dr inż. Marek HOŁYŃSKI, mgr inż. Stanisław JASKÓLSKI, Władysław KLEPACZ (zast. redaktora naczelnego), mgr Andrzej KLIMEK, mgr Stanisław MROZIK, dr inż. Tomasz PAWLAK, doc. dr inż. Andrzej TARGOWSKI
Sekretarz redakcji: Anna GLUTH-NOWOWIEJSKA Red. techn. Ewa SAPOK

RADA PROGRAMOWA

Prof. dr hab. Tadeusz PECHE (przewodniczący), mgr inż. Tomasz BAŃKOWSKI (sekretarz), mgr inż. Antoni BOSSOWSKI, mgr inż. Roman BURNO, doc. dr hab. Andrzej JANICKI, mgr inż. Jan KRAMARCZUK, prof. dr hab. inż. Juliusz KULIKOWSKI, prof. dr Leon ŁUKASZEWICZ, gen. dr inż. Marian PASTERNAK, mgr inż. Bronisław PIWOWAR, mgr Zbigniew SUBSTYK, mgr Jerzy TRYBULSKI, doc. dr hab. Tadeusz WALCZAK, dr inż. Jan ŻYDOWO

Redakcja: 00-011 Warszawa, ul. Jasna 14/16, pokój 331, tel. 27-71-40 lub centrala 26-82-61 w. 285, dyżury redakcji 10,00—13,00

Zakł. Graf. „Tamka". Zam. 252. Papier druk. sat. IV kl. 70 g. A1. Obj. 7,5 ark. druk. Nakład 7000. S-23.

Cena egzemplarza zł 25.— INDEKS 36210/36124 Prenumerata roczna zł 300.—

Informatyka

zastosowania w gospodarce, technice i nauce

Nr 7-8
MIESIĘCZNIK
1978
ROK XIII
lipiec — sierpień

ORGAN KOMITETU INFORMATYKI, MINISTERSTWA NAUKI, SZKOLNICTWA WYŻSZEGO I TECHNIKI ORAZ KOMITETU NAUKOWO-TECHNICZNEGO NOT DS. INFORMATYKI

P.1877/78

BOLESŁAW GLIKSMAN
ZETO Katowice

Komputery JS EMC w ZETO Katowice — doświadczenia eksploatacyjne (część 1)

Sprzęt i oprogramowanie podstawowe

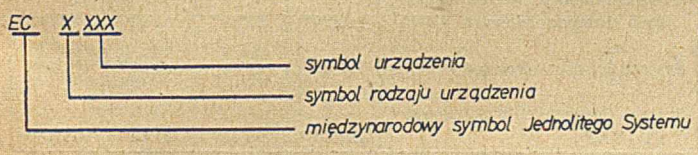
W latach sześćdziesiątych w europejskich krajach socjalistycznych nie istniała wielkoseryjna produkcja systemów komputerowych. W kilkunastu zakładach wytwarzano w niewielkich seriach 24 różne typy komputerów, które były wzajemnie niekompatybilne. Do współpracy z tymi komputerami stosowano ok. 400 różnych urządzeń peryferyjnych, a wymienność oprogramowania istniała tylko w minimalnym zakresie.

Zasadniczą poprawa sytuacji w krajach obozu socjalistycznego nastąpiła na początku lat siedemdziesiątych wraz z podjęciem produkcji i rozwojem zastosowań komputerów Jednolitego Systemu EMC, zwanego potocznie serią RIAD. Skonstruowano i wyprodukowano dotychczas 10 typów jednostek centralnych oraz 120 typów urządzeń peryferyjnych. Przy opracowywaniu i produkcji tego sprzętu zatrudnionych jest łącznie ok. 20 000 naukowców, inżynierów i techników i ok. 300 000 robotników ze 150 instytutów i przedsiębiorstw produkcyjnych. Wzrasta również szybko tempo wzajemnych dostaw sprzętu, które w roku 1975 przekroczyły dziesięciokrotnie wartość osiągniętą w roku 1971.

Wszystkie dotychczas wytworzone jednostki centralne JS EMC należą do tzw. pierwszego pokolenia, wykazującego znaczne podobieństwo z komputerami serii IBM 360. W instytutach naukowych i ośrodkach badawczo-rozwojowych powstaje obecnie tzw. drugie pokolenie serii RIAD, podobne do serii IBM 370.

Łączna liczba zestawów JS EMC eksploatowanych w początkach br. wynosiła 1500. W Polsce mieliśmy w tym czasie w eksploatacji zaledwie 80 zestawów: 25 — typu EC 1020, 8 — EC 1022, 45 — EC 1032, 1 — EC 1040, 1 — EC 1050; w ZETO Katowice — 6 zestawów: 4 — EC 1020, 1 — EC 1022 i 1 — EC 1050.

Celem zorientowania Czytelników w nomenklaturze JS EMC na rysunku poniżej podano sposób kodowania urządzeń.



Sposób kodowania urządzeń JS EMC

Symbole rodzajów urządzeń:
1 — modele komputerów
2 — jednostki centralne
3 — pamięci operacyjne
4 — kanały
5 — pamięci zewnętrzne
6 — urządzenia wejścia
7 — urządzenia wyjścia oraz urządzenia we/wy
8 — urządzenia zdalnego przetwarzania
9 — urządzenia do rejestrowania i przygotowania danych.

PORÓWNANIA PRZEBIEGÓW OBLICZENIOWYCH

Tabele obrazujące parametry techniczne jednostek centralnych i urządzeń zewnętrznych JS EMC były już niejednokrotnie zamieszczane w INFORMATYCE. Jednolity System EMC podlega jednak modyfikacjom i nieustannym badaniom i z tego względu w tabeli 1 zestawiono całość aktualnych danych wszystkich jednostek centralnych.

Z tabeli wynikają dość znaczne różnice w szybkości operacji poszczególnych modeli jednostek centralnych; użytkowników interesują przede wszystkim różnice w zachowaniu się zestawów złożonych z różnych jednostek centralnych i urządzeń zewnętrznych, zwłaszcza przy realizacji zadań z zakresu przetwarzania danych. Wiadome jest, że efektywność zestawu dla tego typu prac w większej mierze zależy od pojemności pamięci oraz rodzaju urządzeń zewnętrznych niż od szybkości arytmometru. Podane w tabeli szybkości operacji zostały zmierzone na podstawie teoretycznie zestawionych mieszanek rozkazów lub problemów. Mieszanka JS jest typową mieszanką rozkazów i służy do odzwierciedlenia przebiegów przy realizacji zadań naukowo-technicznych. Mieszanki Gibson III-E i III-D są mieszankami problemowymi i służą również do oceny przebiegów przy rozbudowywaniu zadań naukowo-technicznych. Mieszanka GPO-WU II jest natomiast typową mieszanką do pomiaru zadań administracyjno-ekonomicznych. Dane zamieszczone w tabeli I świadczą o przydatności modelu JS-1022 do przetwarzania danych ze średnimi prędkościami w szerokim zakresie zastosowań, natomiast szybsze modele — 1032 i 1050 — mogą rozwiązywać zadania naukowo-techniczne z większą sprawnością niż zadania administracyjno-ekonomiczne.

Porównanie szybkości działania komputerów JS w przetwarzaniu danych, przeprowadzone w ZETO Katowice na przebiegach wybranych systemów, prowadzi do wniosku,

Tabela 1. Dane techniczne jednostek centralnych JS EMC

JEDNOSTKI CENTRALNE	2010	2012	2020	2021	2022	2030	2032	2033	2040	2050
Kraj producenta	WRL	WRL	BRL/ZSRR	CSRS	ZSRR	ZSRR	PRL	ZSRR	NRD	ZSRR
Szybkość operacji										
— Mieszanka JS			9 000		91 000	60 000	230 000	200 000	380 000	500 000
— GIBSON III-E			19 000		125 000	70 000	303 000	200 000	411 000	500 000
— GIBSON III-D			10 000		92 000	45 000	212 000	150 000	375 000	408 000
— GPO-WU II			24 000		95 000	38 000	176 000	130 000	240 000	192 000
Liczba rozkazów	86	109	143	71	143	143	143	143	143	143
Pamięć mikroprogramów										
— pojemność (słów)	2048	8192	8192	3072	8192	4096	2816	2048	3072	sterowanie
— czas cyklu (ns)	300	400	1000	250	500	500	320	300	450	układowe
— długość słowa (bitów)	16	16	64	72	64	72	86	128	130	
Czas operacji arytmetycznych										
— dodawanie w st. przecinku	2,1—3,1	2,1—2,9	20,1—33,4	20,3—30,1	2,9—6,5	8,1—13,2	2,1—2,6	1,5—2,0	1,6—1,9	0,7—1,9
— dodawanie w zm. przecinku			57,3—70,8		13,2—15,5	12,8—16,3	4,0—5,0	4,5—5,0	2,8—3,7	1,0—1,9
— mnożenie w st. przecinku	8,0—9,0	8,0—9,0	338—349	80—120	23,0—27,6	9,6—14,1	18,5—19,3	10,0—11,0	6,7—7,6	2,0—2,2
— mnożenie w zm. przecinku			411—413		33,9—36,3	31,9—37,5	13,1—13,2	8,5—9,0	6,0—6,4	1,8—2,1
— dzielenie w st. przecinku	9,0—10,0	9,0—10,0	389—398	145—155	62,7—65,0	108—112	17,5—17,7	14,5—15,2	10,6—11,4	8,2—8,5
— dzielenie w zm. przecinku			389—399		50,2—52,4	56,4—58,7	13,8—14,5	12,5—13,2	7,6—8,4	5,8—5,9
— krótkie operacje	2,2—3,2	2,2—3,0	10,2 -2- 25,3	14,3—15,4	4,1—8,6	10,2—19,3	2,5—4,0	1,5—2,0	2,4—3,5	0,8—1,7
Pamięć operacyjna										
— pojemność (k bajt)	8—64	8—64	64—256	16—64	128—512	128—512	256—1024	512	256—1024	256—1024
— czas cyklu (μs)	0,8	1,0	2,0	2,0	2,0	1,25	1,20	1,25	1,35	1,25
— czas dostępu (μs)	0,4	0,5	1,0	1,0	1,0	0,75	0,66	0,75	0,45	0,95
— szerokość dostępu (bajt)	2	2	4	1	4	4	4	4	8	8
— maks. wymiana (M bajt/s)	2,5	2,0	2,0	0,5	2,0	3,2	3,3	3,2	17,7	8,5
Kanał selektorowy	kanał specjalny									
— maks. liczba			2	2	2	3	3	3	6	6
— maks. szybkość (K bajt/s)			300	236	500	800	1030	800	1300	1250
Kanał multipleksorowy	kanał specjalny z adapterem									
— maks. liczba			1	1	1	1	1	1	1	1
— maks. szybkość (K bajt/s)	40	45	20	27	50	40	110	70	50	110

że ekonomicznie uzasadnione wykorzystanie ich mocy obliczeniowej może nastąpić po uzupełnieniu zestawów o duże dyski i urządzenia do zdalnego dostępu. Komputery JS, wnosząc do naszych ośrodków obliczeniowych filozofię dyskową, zmuszają do istotnej reorientacji podejścia projektantów i programistów oraz użytkowników. Systemy za-

Tabela 2. Porównanie czasów przebiegów programów systemu gospodarki materiałowej SYGMAT w komputerach o konfiguracji taśmowej (MIŃSK-32) i dyskowej (EC 1020 i EC 1022)

Nazwa programów	Czas przebiegu w komputerach w minutach		
	MIŃSK 32	EC 1020	EC 1022
Tworzenie zbiorów roboczych oraz aktualizacja kartoteki	60	66	44
Sortowanie zbiorów	75	17	14
Emitowanie wydawnictw	144	78	27
System SYGMAT łącznie	279	161	85

projektowane i eksploatowane na komputerach w konfiguracji taśmowej (np. M-32) przebiegały dłużej niż na komputerach JS, nawet w przypadku modeli o wolniejszej jednostce centralnej (EC 2020) — patrz tabela 2.

Przeniesienie systemów na maszyny JS nie daje jeszcze spodziewanych znacznych redukcji czasów przebiegu, ponieważ obarczone jest błędami starej technologii, co powoduje, że możliwości tych maszyn nie są jeszcze w pełni wykorzystane. Dotyczy to w szczególności pozostawienia organizacji zbiorów na taśmach magnetycznych, a w konsekwencji zachowania dotychczasowych metod przetwarzania.

W tabeli 3 przedstawiono czasy przebiegu pakietów programów różnych systemów przetwarzania danych. Przebiegi te zostały zrealizowane przez tych samych operatorów pod kontrolą systemu operacyjnego DOS. Fakt ten ma istotne znaczenie ze względu na konieczność stworzenia warunków podobnych przy porównywaniu. Jak wiadomo bowiem, czas reakcji operatora na instrukcję systemu operacyjnego bardzo rzutuje na uzyskane wyniki przy porównaniu zwłaszcza czasu krótkich przebiegów. Mimo takich założeń nie udało się zapewnić pełnej porównywalności, która mogłaby nastąpić dopiero przy maksymalnym, wieloprogramowym działaniu komputerów pod kontrolą systemu operacyjnego OS. Warunków takich nie można było stworzyć, ponieważ posiadane konfiguracje badanych modeli nie pozwalają na taki rodzaj eksploatacji.

We wszystkich mierzonych przebiegach zapisywane były czasy wykonywania poszczególnych podprogramów. Analiza tych czasów pozwala stwierdzić, że szybkości operacji polegających na aktualizacji i sortowaniu danych są zbliżone dla modeli EC 1032 i EC 1050 oraz mniejsze dla jednostek EC 1020 (o 50%) i EC 1022 (o 20%), natomiast szybkości operacji wejścia i wyjścia są bardziej zróżnicowane w zależności od tematu i większe w modelach o wyższych numerach kodowych.

Tabela 3. Porównanie czasów przebiegu pakietów programów różnych systemów przetwarzania danych w komputerach JS

Nazwa systemu	Czas przebiegu w komputerach			
	EC 1020	EC 1022	EC 1032	EC 1050
System ewidencji sald na rachunkach oszczędnościowych PKO	113'20"	53'39"	34'22"	39'14"
System finansowo-kosztowy	56'46"	41'31"	43'25"	37'08"
System Gospodarki Materiałowej SYGMAT	122'40"	93'41"	74'34"	77'13"

Rozpatrując zagadnienia występujące przy eksploatacji systemów EPD na maszynach Jednolitego Systemu rozróżnia się dwa czynniki decydujące o efektywności tych systemów. Pierwszy czynnik związany jest z opanowaniem przez pracowników obsługi konserwacji i napraw komputerów o odmiennej konstrukcji i technologii, jak również posługiwaniem się systemami eksploatacji opartymi na innej filozofii. Drugim czynnikiem decydującym o efektywności systemów EPD są walory techniczne (niezawodność działania) komputera.

Biorąc pod uwagę obydwa czynniki, wyniki uzyskane podczas użytkowania maszyn Jednolitego Systemu przez okres co najmniej jednego roku potwierdzają powszechnie znaną prawdę, że od kwalifikacji i doświadczenia personelu ośrodka zależą jakość pracy i stopień wykorzystania komputerów.

Ilustrują to znakomicie dane zebrane w Zjednoczeniu Informatyki i przedstawione w tabeli 4. Odnoszą się one do procentowego wykonania usług obliczeniowych w stosunku do rocznego normatywu czasu pracy komputerów Jednolitego Systemu w okresie dochodzenia do pełnej zdolności eksploatacyjnej.

Tabela 4. Procentowe wykonanie usług obliczeniowych w stosunku do rocznego normatywu czasu pracy komputerów JS EMC

Komputer	Normatyw	Średnia ZI	Odchylenie
EC 1020	52	36	—16
EC 1022	24	32	+8
EC 1032	13	16	+3
EC 1050	13	13	0

Z tabeli tej wynika niedwuznacznie, że są ośrodki obliczeniowe, które już dobrze opanowały posługiwanie się sprzętem JS. Świadczy to o dobrej jakości sprzętu i o wystarczającym oprogramowaniu.

WŁASNOŚCI TECHNICZNO-EKSPLOATACYJNE

Dane dotyczące czasów międzyawaryjnych wybranych egzemplarzy komputerów JS w pierwszym i drugim roku eksploatacji w woj. katowickim, zamieszczone w tabeli 5,

Tabela 5. Wskaźniki awaryjności wybranych egzemplarzy komputerów JS w pierwszym i drugim roku eksploatacji (w godzinach)

EC 1020 nr fabr. 80M

Miesiąc		III 76	IV 76	V 76	VI 76	VII 76	VIII 76	IX 76	X 76	XI 76	XII 76	I 77	II 77
Łączny czas pracy	Tp	138	186	193	241	222	232	230	332	360	366	499	494
Łączny czas awarii	Ta	68,6	107,6	22,0	95,9	39,3	35,0	26,2	24,8	47,2	22,1	77,9	32,2
Liczba przestojów awaryjnych	Ia	11	20	16	24	11	12	8	6	17	14	15	25
Czas międzyawaryjny	Tm	6,3	3,9	10,7	6,1	16,6	16,4	25,5	33,0	18,4	24,6	24,7	18,5

EC 1020, nr fabr 80M

Miesiąc		III 77	IV 77	V 77	VI 77	VII 77	VIII 77	IX 77	X 77	XI 77	XII 77	I 78	II 78
Łączny czas pracy	Tp	460	348	439	454	432	447	444	500	492	536	559	428
Łączny czas awarii	Ta	79,6	45,0	2,2	1,7	18,2	13,9	43,3	15,3	7,7	6,7	5,9	19,4
Liczba przestojów awaryjnych	Ia	25	6	2	3	8	7	12	7	6	5	8	11
Czas międzyawaryjny	Tm	15,2	50,5	218,4	150,8	51,7	61,9	33,4	69,2	80,7	105,9	69,1	37,2

EC 1022, nr fabr. 35M

Miesiąc		III 77	IV 77	V 77	VI 77	VII 77	VIII 77	IX 77	X 77	XI 77	XII 77	I 78	II 78
Łączny czas pracy	Tp	404	452	435	376	374	423	431	445	398	472	527	618
Łączny czas awarii	Ta	63,5	71,2	38,1	33,0	7,5	47	77,3	—	23,1	32,0	41	35,2
Liczba przestojów awaryjnych	Ia	13	10	14	10	7	16	13	—	6	10	7	8
Czas międzyawaryjny	Tm	26,2	38	28,3	34,3	53	23,5	27,2	—	62,5	44,0	69,4	73

EC 1032, nr fabr. 023

Miesiąc			IV 77	V 77	VI 77	VII 77	VIII 77	IX 77	X 77	XI 77	XII 77	I 78	II 78
Łączny czas pracy	Tp		68	178	197	193	240	217	317	313	205	258	202
Łączny czas awarii	Ta		15	34	74	20	81	20	24	29	5	49	8
Liczba przestojów awaryjnych	Ia		10	13	17	6	19	24	17	21	16	29	11
Czas międzyawaryjny	Tm		5,3	11,1	7,2	29	8,4	10,4	17,2	13,5	12,5	7,2	17,6

EC 1050, nr fabr. 6005

Miesiąc					VI 77	VII 77	VIII 77	IX 77	X 77	XI 77	XII 77	I 78	II 78
Łączny czas pracy	Tp				161,2	282	368	396	414	421	551	545	530
Łączny czas awarii	Ta				58,4	65	50,3	41,5	69,1	174,7	61	33,3	18
Liczba przestojów awaryjnych	Ia				37	45	50	59	20	20	24	21	18
Czas międzyawaryjny	Tm				2,8	4,8	6,3	6,0	17,2	12,3	20,4	24,3	23 6

świadczą o pewnej analogii w dochodzeniu do coraz lepszych rezultatów. Wszystkie komputery JS po opanowaniu problemów eksploatacyjnych przez personel techniczny i po wyeliminowaniu słabszych elementów osiągają po roku przebiegi międzyawaryjne rzędu 20 godzin. Zaznaczyć należy, że model EC 1022 (ulepszony model EC 1020) wykazuje znacznie lepsze wyniki osiągając poziom 40 godzin pracy międzyawaryjnej.

Analiza powtarzających się błędów lub awarii wskazuje na daleko idące podobieństwa w zachowaniu się wszystkich modeli Jednolitego Systemu. Oto spostrzeżenia personelu technicznego.

W przypadku wystąpienia błędu maszynowego dotyczącego procesora lub pamięci operacyjnej wykonywano następujące czynności:
— odnotowywano stany odpowiednich rejestrów i wskaźników
— dokonywano zapisu w obszar diagnostyczny pamięci, stanów rejestrów procesora lub kanałów
— podejmowano próbę restartu.

W przypadku niemożności kontynuowania dalszej pracy pod nadzorem systemu operacyjnego (błąd maszynowy nie przemijający) przystępowano do lokalizacji awarii.

Przy niewłaściwej pracy kanałów selektorowego bądź multipleksorowego postępowano w sposób następujący:
— w przypadku zawieszeń sprawdzano stan wskaźników na pulpicie lokalnym urządzenia zewnętrznego oraz przeprowadzano analizę prawidłowości stanu; następnie podejmowano próbę restartu oraz w przypadku niepowodzenia ładowano ponownie system operacyjny
— jeśli występowały błędy wykryte przez kanał, sprawdzano autonomicznie poprawność pracy.

W przypadku wystąpienia błędu stałego przystępowano do lokalizacji awarii.

Typowe przyczyny błędów występujących w procesorze, pamięci operacyjnej i kanałach były następujące:
— zły styk
— zwarcie lub przerwa w przewodach
— zniszczony element elektroniczny na pakiecie.

W przypadku awarii urządzenia zewnętrznego, urządzenie odłączano i przystępowano do jego naprawy poza systemem. W większości przypadków awaria tego typu nie powoduje przestoju maszyny ze względu na istnienie urządzeń rezerwowych.

SYSTEM EKSPLOATACJI

Każdy użytkownik komputera powinien stworzyć własny, dostosowany do swoich potrzeb system eksploatacji, wychodząc ze zbioru modułów systemu dostarczonych przez producenta. Niezbędnym modułem systemu eksploatacji są programy sterujące, nazywane powszechnie systemami operacyjnymi. Wybór modułów zależy od konfiguracji komputera oraz od rodzaju wykonywanej pracy.

Konstruktorzy komputerów Jednolitego Systemu przyjęli jako podstawowe moduły systemu eksploatacji systemy operacyjne DOS/JS i OS/JS.

W Polsce dostępne są trzy wersje systemu DOS/JS. Różnią się one między sobą wielkością bibliotek, liczbą dostępnych translatorów oraz rodzajem dostępu. DOS/JS zawiera translatory języków: ASSEMBLER, FORTRAN, BASIC, FORTRAN IV, COBOL, PL/1, RPG. DOS wprowadza sztywną organizację pamięci operacyjnej z podziałem na 3 partycje (strefy) oraz wymaga przydziału do każdej partycji określonych urządzeń. Wyższe wersje DOS/JS dysponują dodatkowymi środkami, umożliwiającymi lepsze wykorzystanie komputera. Należy do nich w szczególności program wspomagający, który rezyduje w jednej z trzech partycji i realizuje funkcje wejścia oraz wyjścia. Zapewnia on równomierne obciążenie zadaniami wszystkich partycji za pomocą jednego urządzenia wejściowego i jednego urządzenia wyjściowego. DOS/JS z tym programem może być stosowany w następujących warunkach:
— optymalna wielkość programu SUPERVISOR — 24 K bajtów (minimalna ok. 10 K bajtów)
— optymalna wielkość programu wspomagającego — 38 K bajtów
— system mieści się na 1 pakiecie dysków magnetycznych o pojemności 7,25 MB
— korzystanie z programu wspomagającego wymaga dodatkowego pakietu dysków magnetycznych
— system pracuje efektywnie nawet na stosunkowo wolnych modelach (EC 1020).

Tabela 6. Niezbędna konfiguracja sprzętu dla systemów operacyjnych DOS/JS i OS/JS

	DOS/JS	OS/JS		
		PCP	MFT	MVT
Pamięć operacyjna [pojemność K bajtów]	64	128	256	512
Pamięć dyskowa [jednostek]	1	2—4	4—8	⩾8
Pamięć taśmowa [jednostek]		2—4	4—8	⩾8
Czytnik kart [liczba]	1	1	2	2
Perforator kart [liczba]	1	1	1	1
Drukarka wierszowa [liczba]	1	1	2	3
Monitor operatora [liczba]	1	1	1	1

System OS/JS składa się z dwu grup programów: programów sterujących i programów przetwarzających. Istnieją trzy typy programów sterujących: program PCP, nadzorujący wykonanie jednego zadania; program MFT, nadzorujący wykonanie stałej liczby zadań; program MVT, nadzorujący wykonanie zmiennej liczby zadań. System OS/JS zawiera translatory języków wymienione przy charakterystyce systemu DOS/JS, jednak o szerszym repertuarze rozkazów. Poza translatorami OS/JS zawiera różne programy usługowe, znacznie rozszerzające jego zastosowanie w porównaniu do DOS/JS. W niektórych krajowych ośrodkach obliczeniowych stosuje się z powodzeniem wersję OS MVT.

OS MVT można stosować w następujących warunkach:
— optymalna wielkość programu SUPERVISOR — 180 K bajtów (minimalna ok. 140 K bajtów)
— system mieści się na 2 pakietach dysków magnetycznych o pojemności po 7,25 MB
— korzystanie z systemowego wyjścia wymaga dodatkowego pakietu dysków magnetycznych
— system pracuje efektywnie przy prędkościach jednostki centralnej 100 tys. operacji/s.

Tabela 7. Podstawowe własności systemów DOS/JS i OS/JS

DOS/JS	OS/JS
● Stały podział pamięci operacyjnej dla maksimum 3 równolegle wykonywanych programów	● Dynamiczny podział pamięci operacyjnej dla maksimum 15 równolegle wykonywanych programów
● Gotowe do realizacji programy użytkowe są przypisane do konkretnych adresów co uniemożliwia ich realizację w dowolnym obszarze pamięci operacyjnej	● Programy użytkowe mogą być realizowane w dowolnym fragmencie pamięci operacyjnej (programy samoprzemieszczalne)
● Programy są przypisane do konkretnych urządzeń WE/WY (zależność programów od WE/WY)	● Programy są niezależne od urządzeń WE/WY (zbiory i urządzenia WE/WY są definiowane na etapie realizacji programu)
● Istnieje możliwość stosowania namiastki procedur, polegającej na zapisaniu na dysk szeregu kart z operatorami sterującymi	● Zapewnia stosowanie sparametryzowanych procedur, co w sposób zasadniczy usprawnia sposób przygotowania zadań do realizacji
	● Istnieje systemowy katalog zbiorów (system zapewnia informację i dotarcie do utworzonych zbiorów)
	● Umożliwia tworzenie kodów zakończenia kroków zadania i ich badanie; wynik badania przesądza o sposobie realizacji zadania
	● System zapewnia automatyczne rozpoznanie tomów
	● Umożliwia pracę w trybie *time sharing*
	● Gwarantuje automatyczny restart
	● Zawiera telekomunikacyjną metodę dostępu

W zamieszczonym poniżej artykule przedstawiamy propozycję transliteracji alfabetu rosyjskiego na łaciński, niezbędnej w zautomatyzowanych systemach informacji naukowej, technicznej i ekonomicznej (inte). Zachęcamy Czytelników do dyskusji nad tym rozwiązaniem.

STEFAN J. NIEMENTOWSKI
Instytut Energetyki, Branżowy Ośrodek
Informacji Naukowej, Technicznej i Ekonomicznej
Warszawa

Transliteracja alfabetu rosyjskiego na łaciński w systemach informatycznych

Artykuł niniejszy przedstawia nowy, optymalny pod wieloma względami sposób transliteracji, któremu autor nadał nazwę systemu ESEN. Potrzeba opracowania takiego systemu wynikła stąd, że żaden z istniejących w naszym kraju nie nadaje się w pełni na potrzeby epd.

W praktyce epd dość często występują teksty lub elementy tekstu (wyrazy, skrótowce) obcojęzyczne, w tym także rosyjskie. Ponieważ stosowane w komputerach drukarki najczęściej nie mogą wypisywać tekstów alfabetem rosyjskim [1]), dokonuje się transliteracji lub transkrypcji na alfabet łaciński [2]).

Transliteracja polega na zastępowaniu liter jednego alfabetu literami innego alfabetu (ewentualnie uzupełnionymi znakami diakrytycznymi) bez konieczności uwzględniania fonetyki głosek odpowiadających literom zastępowanym. Stosuje się więc system „litera za literę", tj. stale zastępuje się jedną literę określoną inną literą lub zespołem liter. System ten jest dość prosty i z tego powodu często stosowany. Zapewnia on również łatwość i pewność retransliteracji, tj. powrotu do alfabetu (tekstu) oryginalnego.

Transkrypcja stara się przy zastępowaniu liter oddawać też fonetykę głosek zastępowanych, dlatego jest dużo trudniejsza i z tych względów mniej rozpowszechniona. Daje jednak wierniejszy, bardziej naturalny w czytaniu obraz tekstu oryginalnego. Retranskrypcja jest nader kłopotliwa, nie może się odbywać „mechanicznie" i wymaga dokładnej znajomości pisowni wyrazów oryginalnych.

Zarówno transliteracja, jak i transkrypcja dokonywane w Polsce mają jednak poważną wadę przy stosowaniu ich w epd, gdyż zawierają znaki diakrytyczne, których nie mogą drukować komputery. W myśl obowiązującej w Polsce transliteracji wg PN-70/N-01201 [4], np. litery cyrylickie ч, ш, ж należy transliterować jako č, š, ž, natomiast komputery drukują je w formie C, S, Z (tj. bez znaków diakrytycznych), zniekształcając w ten sposób wyrazy rosyjskie i utrudniając ich wierną i szybką retransliterację. Zjawisko to nasila się w przypadku nazw własnych, nazwisk, skrótów itp. i wówczas nie można domyślić się z całą pewnością oryginalnego brzmienia: np. nazwisko Кошматский będzie wydrukowane przez komputer jako KOSMATSKIJ, a wówczas nie wiadomo, czy w oryginale było Кошматский czy Космatский. Podobnie wydrukowany przez komputer skrótowiec PZS moe oznaczać: ПЗС, ПЖЩ, ПЗШ lub ПЖС w tekście oryginalnym rosyjskim.

Przedstawiony tutaj po raz pierwszy system transliteracji ESEN (nazwa utworzona z inicjałów imienia i nazwiska autora systemu i niniejszego artykułu) usuwa powyższe wady i pozwala na szybkie i dokładne odczytywanie drukowanych przez komputery transliterowanych tekstów rosyjskich, a także łatwe i bezbłędne dokonywanie ich retransliteracji na język oryginału. System ten oczywiście przedstawiony jest jako propozycja do dyskusji, gdyż trudne i skomplikowane zagadnienie transliteracji na potrzeby epd nie jest dotychczas rozwiązane zadowalająco nie tylko w skali krajowej, ale — przede wszystkim — międzynarodowej, o czym będzie jeszcze mowa.

Zawężenie zagadnienia do współczesnego alfabetu rosyjskiego jest świadome i uzasadnione tym, że w praktyce polskiej zetknięcie się przy epd z tekstami cyrylickimi w innym języku niż współczesny rosyjski jest bardzo rzadkie. Nie wyklucza to jednak perspektywicznych prac nad uzupełnieniem i rozszerzeniem systemu ESEN na inne języki posługujące się cyrylicą: ukraiński, białoruski, bułgarski, serbochorwacki, macedoński itd.

---

1) Alfabet rosyjski — inaczej alfabet cyrylicki (cyrylica); terminu tego jednak rzadko się tutaj używa, gdyż niniejsza praca dotyczy wyłącznie współczesnych tekstów rosyjskich, a więc z pominięciem tych liter cyrylicy, które we współczesnym alfabecie rosyjskim nie są stosowane.

2) Alfabet łaciński, zwany też „rzymskim", przy czym chodzi tu o jego wersję 26-literową, tzn. uzupełnioną o litery „j" i „w", nie znane łacinie klasycznej (z jej „złotego okresu"). Wersja ta nosi różne nazwy: alfabetu łacińskiego modyfikowanego [1], współczesnego [2], uzupełnionego (w średniowieczu) [3]. Wskazane byłoby ustalenie przez językoznawców jednego, ogólnie przyjętego terminu dla tej wersji.

Inż. STEFAN J. NIEMENTOWSKI ukończył studia w 1952 r. na Wydziale Elektrycznym Politechniki Warszawskiej i w tym samym roku podjął pracę jako konstruktor i technolog maszyn elektrycznych. Następnie pracował w dziedzinie normalizacji elektroenergetyki. Od 1972 r. jest adiunktem naukowo-badawczym w Instytucie Energetyki i pracuje w Branżowym Ośrodku Informacji Naukowej, Technicznej i Ekonomicznej, m. in. przy wdrażaniu i eksploatacji skomputeryzowanego systemu informacyjno-wyszukiwawczego inte w dziedzinie elektroenergetyki — ASIWELEN.

MOŻLIWOŚCI DRUKU CYRYLICĄ PRZEZ KOMPUTERY

Przy rozpatrywaniu możliwości — a także i celowości — posługiwania się cyrylicą, w całym cyklu epd należy wyodrębnić trzy zagadnienia: bazy komputerowej, bazy urządzeń do przygotowania danych oraz posługiwania się wydrukami.

W zakresie bazy komputerowej, m. in. drukarek współpracujących z komputerami, do połowy 1977 r. nie było możliwe w Polsce drukowanie tekstów alfabetem rosyjskim, gdyż drukarki zarówno połączone z komputerami serii ODRA 1300 [5], jak i z innymi komputerami nie mają dodatkowych znaków dla cyrylicy, a komputery odpowiedniego kodu wewnętrznego. Np. w komputerach ODRA i ICL kod jest ograniczony do 64 znaków, obejmujących 26 dużych liter łacińskich (tzw. wersalików), 10 cyfr arabskich oraz 28 znaków pomocniczych (interpunkcyjnych, matematycznych itp.). Dopiero od niedawna, w miarę uruchamiania w kraju komputerów Jednolitego Systemu (RIAD), powstały możliwości wypisywania tekstów z użyciem alfabetu rosyjskiego na drukarkach tego systemu, np. EC-7033 [6], które w swym repertuarze znaków mają duże litery łacińskie i cyrylickie. Komputery RIAD w kodzie wewnętrznym DKOI przewidują dla drukarek wierszowych bębnowych (standardowych) 90 znaków, natomiast dla drukarek łańcuchowych [3]) — 136 znaków. W tym drugim przypadku do repertuaru znaków dochodzą małe litery: 26 łacińskich i 20 cyrylickich. Jeśli jednak wziąć pod uwagę bardzo małe jeszcze rozpowszechnienie w kraju komputerów RIAD, można sformułować tezę, że druk cyrylicą przez komputery w Polsce jest na razie b. trudny do zrealizowania.

Krajowa baza urządzeń do przygotowania danych cechuje się małą stosunkowo liczebnością maszyn do pisania z alfabetem rosyjskim, a jeszcze mniejszą — prawie znikomą — liczbą automatów organizacyjnych, dziurkarek i czytników mogących wytwarzać lub akceptować teksty zapisywane cyrylicą na maszynowych nośnikach informacji. Należy też uwzględnić bardzo szczupłe kadry maszynistek i perforatorek władających zadowalająco rosyjskim i umiejących biegle posługiwać się odpowiednim dla tego języka sprzętem. Ponadto — ponieważ teksty rosyjskie (np. w informacji naukowej przy sporządzaniu kart dokumentacyjnych) są zazwyczaj tylko fragmentem tekstu polskiego (lub innego tekstu pisanego alfabetem łacińskim) — występuje konieczność przerzucania opracowań z maszyny do pisania lub automatu organizacyjnego z czcionkami łacińskimi na maszynę do pisania lub automat organizacyjny z alfabetem rosyjskim.

Uzyskiwanie wydruków pisanych alfabetem rosyjskim jest — jak to już wyjaśniono — trudne, kosztowne i pracochłonne. Poza tym istnieje duże niebezpieczeństwo pomylenia tekstu oryginalnego rosyjskiego z polskim lub innym, pisanym alfabetem łacińskim, np. z tekstem angielskim. Ma to miejsce w przypadku krótkich, rozsianych fragmentów tekstu rosyjskiego, zawierającego litery cyrylickie o kształcie zgodnym z literami łacińskimi. Przykładem może być tekst: „ZALECENIE PC 514-75 i NORMA CT 226-75 POSTANAWIAJA..." — w którym znajdujące się przed liczbami litery rosyjskie o brzmieniu (w transliteracji) „RS" (Rekomendacija po Standartizacii) i „ST" (Standart) łatwo mogą być wzięte za litery łacińskie „PC" i „CT".

Jak wynika z powyższych rozważań, w każdym z wymienionych zagadnień odpowiadających trzem etapom procesu epd najdogodniejsze w naszym kraju jest stosowanie alfabetu łacińskiego, a więc w przypadku tekstu rosyjskiego — przetransliterowanie go na alfabet łaciński i dopiero w takiej postaci przekazywanie do przetwarzania na komputerze.

STOSOWANE SYSTEMY TRANSLITERACJI I TRANSKRYPCJI

System ISO (podstawowy) z 1968 r., ujęty w zaleceniu ISO/R 9-1968 [7], charakteryzuje się częstym używaniem liter ze znakami diakrytycznymi (np. šč dla rosyjskiego щ, é dla э itd). Dopuścił on (m. in. na skutek starań Polski) oddawanie rosyjskiego x przez ch (w tabeli głównej przyjęto h) i przepisuje — wbrew tradycjom anglo-amerykańskim — użycie w wielu przypadkach litery j (dla й, a także w znakach dwuliterowych: ja dla я i ju dla ю). System ten stał się podstawą do wydania norm transliteracji w wielu krajach (zazwyczaj jednak z małymi odstępstwami), np. w Rumunii, we Francji. Część krajów oparła swe normy na poprzedniej wersji zalecenia z 1954 r. (ISO/R 9-1954), np. Węgry, Szwecja, Dania, Belgia. Oczywiście — z powodu użycia znaków diakrytycznych — podstawowy system ISO nie może być zastosowany w epd.

System ISO zmodyfikowany (tzw. „anglo-amerykański") zawarty jest w zaleceniu ISO/R 9-1968 [7] jako dopuszczalny wariant dla krajów, w których już tradycyjnie ugruntował się. Zawiera on m. in. znak diakrytyczny (gdyż й oddawane jest przez i) oraz znak dla ë. Transliteracja wg tego systemu brzmi jednak dla Polaka obco, np. ч oddawane przez ch, x przez kh, щ jako shch itd. Omawiany tu zmodyfikowany system ISO może być — kosztem pominięcia nielicznych jego znaków diakrytycznych — stosowany w epd, jednak opuszczenie tych znaków powoduje zniekształcenie i niejednoznaczność transliteracji, a ponadto za małe skojarzenia brzmieniowe z językiem polskim, w sumie dyskwalifikują go jako sposób transliteracji dla epd w naszym kraju.

System transliteracji polski ujęty jest w PN-70/N-01201 [4] i opiera się dość dokładnie na wersji podstawowej systemu ISO, od której różni się paroma szczegółami. Z uwagi na częste stosowanie znaków diakrytycznych, system polski, podobnie jak podstawowy system transliteracji ISO, nie może być używany w epd.

System transliteracji dla dalekopisów opiera się w Polsce na regulaminie telegraficznym [8] i dotyczy telegramów oraz dalekopisów. System ten opiera się na mieszanych (francusko-angielsko-niemieckich) zasadach transliteracji bez znaków diakrytycznych i zawiera szereg poważnych niedociągnięć: brak odpowiedników transliteracyjnych dla czterech liter rosyjskich, niekonsekwencja przy transliteracji litery щ, brak znaków rozłączających przy zbiegu liter, jednakowa transliteracja dwóch różnych liter rosyjskich itp. W związku z tymi niedociągnięciami i za małymi skojarzeniami brzmieniowymi z językiem polskim — system ten nie nadaje się do użytku w procesach epd w naszym kraju.

Systemy transliteracji MCNTI [4]) obejmują transliterację tradycyjną oraz transliterację na potrzeby epd, a zawarte są w NTP MCNTI 17-76 [9] z 1976 r. Transliteracja tradycyjna w tym systemie charakteryzuje się używaniem tylko pojedynczych liter jako odpowiedników liter rosyjskich, co wymagało zastosowania aż siedmiu rodzajów znaków diakrytycznych (nie wszystkie są w zestawie znaków maszyn do pisania stosowanych w Polsce). Niektóre transliteracje budzą za małe skojarzenia (np. a dla ros. „twardego znaku", ü i ä dla ros. ю i я). Dla dziesięciu liter rosyjskich dopuszczono, oprócz podanej w tablicy, transliterację odmienną wg norm krajów członkowskich MCNTI (nie wiadomo jednak dokładnie, czy postanowienie to rozciąga się na obydwa systemy transliteracji — tj. tradycyjny i na potrzeby epd — czy tylko na tradycyjny).

Transliteracja na potrzeby epd nie zawiera znaków diakrytycznych i posługuje się najwyżej dwoma literami. Występują tu duże rozbieżności między transliteracją tradycyjną (tr.) — a transliteracją na potrzeby epd (epd). Np. rosyjskie й oddawane jest jako j (tr.) lub jj (epd), rosyjskie ë występuje w formie ë (tr.) i jo (epd). Bardzo małe skojarzenia budzi np. hh (epd) jako ekwiwalent rosyjskiego щ.

Rozbieżności między transliteracją tradycyjną a epd jest bardzo dużo i nie wszystkie wydają się uzasadnione. Np. ju, ja (epd) dla rosyjskich ю, я mogłoby z powodzeniem być użyte również w systemie tradycyjnym (zamiast ü, ä) — tym bardziej, że w „Danych informacyjnych" do wspomnianych zaleceń NTP zamieszczono uwagę o oparciu się na postanowieniach ISO, a przecież transliteracja ju, ja jest zgodna z podstawowym systemem ISO.

Konkludując — obydwa systemy transliteracji MCNTI cechują się wieloma wadami i aczkolwiek drugi z nich może być stosowany w komputerach, jednak związane byłoby to na naszym terenie z wieloma trudnościami. Część tych trudności już tu omówiono, a pozostałe wymienione są przy rozpatrywaniu systemu ISO zmodyfikowanego, do którego wersja dla epd częściowo nawiązuje.

---

[3]) Wykonywane są one na specjalne zamówienie.

[4]) Skrót nazwy Meždunarodnyj Centr Naučnoj i Techničeskoj Informacii — organ działający w ramach RWPG, wydaje m. in przepisy pod nazwą NTP, czyli Normativno-Techničeskoe Predpisanie — rodzaj zaleceń z dziedziny informacji naukowej (niektóre z tych zaleceń są później ustanawiane jako normy RWPG — o symbolu ST SÉV).

System transliteracji radziecki ujęty jest w normie GOST 16876—71 z 1971 r.[5], a oparto go głównie na zmodyfikowanym systemie ISO z domieszką systemu ISO podstawowego. System radziecki — jako jedyny system narodowy — przewiduje (dopuszcza) dodatkową wersję transliteracyjną trzech liter do użytku epd, w której przewidziano jj dla oddania ros. й (zdwojenie j wydaje się zbyteczne), eh dla э oraz — budzące mało skojarzeń — q jako odpowiednik ros. щ. Reszta liter, z wyjątkiem ё, chętnie przez Rosjan przedstawianym jako e — da się oddać transliteracyjnie wg systemu radzieckiego również i w epd. W załączniku do tego GOST podkreślono „tradycyjność" jako jedną z istotnych i pożądanych cech systemu transliteracyjnego w ogólności. Z powodu oparcia się na zmodyfikowanym wariancie ISO, system radziecki nie nadaje się do użytku w naszym kraju na potrzeby epd, jak to już umotywowano poprzednio przy omawianiu systemów ISO.

Systemy transkrypcji polskiej omówione będą tylko z uwagi na przejęcie z nich niektórych zestawów liter do systemu ESEN. Transkrypcja polska biblioteczna ujęta jest w instrukcji CIDNT[6] z 1951 r. [10] (bardzo zbliżona do niej wersja występuje u Jodłowskiego i Taszyckiego [11]). Charakteryzuje się ona oparciem o praktykę językowo-brzmieniową polską, np. ros. ч oddawane jest przez cz itp. Występują tu jednak znaki diakrytyczne w literach „ż" i „ł", jako odpowiednikach transliteracyjnych ros. ж i л. Mimo skomplikowanych reguł transkrypcji system ten jest dość rozpowszechniony w Polsce. Z powodu znaków diakrytycznych i trudnych reguł postępowania — nie nadaje się w całości do celów epd.

Warto jeszcze nadmienić o polskiej transkrypcji wydawniczej [11], która cechuje się pewnymi uproszczeniami w stosunku do transkrypcji bibliotecznej i znacznym spolonizowaniem nazw i nazwisk rosyjskich.

SYSTEM TRANSLITERACJI ESEN

Jak wynika z dokonanego przeglądu - szerzej stosowanych systemów transliteracji — brak wśród nich rozwiązania optymalnego na potrzeby epd.

Próbę takiego rozwiązania przedstawia zamieszczona obok „Tabela transliteracji współczesnego alfabetu rosyjskiego wg systemu ESEN", zawierająca zestawienie 34 liter i znaków rosyjskich oraz ich odpowiedników transliteracyjnych.

Pewnego wyjaśnienia wymaga poruszona w uwadze nr 3 do tej tabeli kwestia małych lub dużych liter, używanych w drukarkach komputerów. Ponieważ drukarki większości komputerów mogą pisać tylko dużymi literami (wersalikami), powstaje jakby mutacja systemu ESEN (ESEN-W), charakteryzująca się częstszym użyciem znaku = (zwłaszcza w przypadku skrótowców) w porównaniu z drukarkami łańcuchowymi, dysponującymi także małymi literami. Dla nielicznych jeszcze u nas drukarek łańcuchowych tworzy się więc druga, bardziej czytelna, bo pozbawiona znaku nieliterowego „=" mutacja systemu ESEN (ESEN-M). Jeśli zakłada się z góry, że użytkownik tabulogramów nie będzie dokładnie zorientowany, z jakiego typu drukarki pochodzą wydruki, a uzyskane teksty nie wskazują na rodzaj drukarki (możliwość małych i dużych lub tylko dużych liter), wówczas bezpieczniej jest stosować mutację ESEN-W, która całkowicie eliminuje ewentualne niejednoznaczności przy retransliteracji skrótowców[7].

System transliteracji ESEN opiera się na trzech podstawowych zasadach:

1) Łatwe i naturalne (tradycyjne) kojarzenie: oparto się na naturalnych w języku polskim zestawach liter (np. „cz", „sz") dla oddania odpowiadających im liter (i tak samo brzmiących głosek) rosyjskich — sposób ten jest od dawna ugruntowany przez system transkrypcji polskiej alfabetu rosyjskiego [10, 11]. Cechą odróżniającą ESEN od powyższego systemu jest jednak przyjęcie zasady transliteracji, pozwalającej ominąć trudności i niedogodności transkrypcji i związanej z nią retranskrypcji.

2) Linearne użycie znaków diakrytycznych: tam, gdzie nie było możliwe pominięcie znaków diakrytycznych, zastosowano je (ograniczając te przypadki do minimum), ale w sposób umożliwiający proste wydrukowanie na komputerze, mianowicie znak diakrytyczny następuje bezpośrednio po literze, której dotyczy, np. lp. 30 — e', lp. 6 — e" (w przeciwieństwie do systemu ISO [7] i PN [4], gdzie był układ „piętrowy", tj. znak diakrytyczny znajdował się nad literą, np. č, è, ë).

3) Duża zgodność z systemem ISO w tych przypadkach, gdy możliwy jest druk przez komputer znaków systemu ISO — zachowano więc litery „v", „l" (wbrew temu, że w transkrypcji polskiej [10, 11] występują „w", „ł"), znak ' (dla rosyjskiego „twardego znaku") oraz „zh" (dla litery ж), co przewiduje ISO jako wariant anglo-amerykańskiego systemu; ponadto w kilku przypadkach nawiązano do systemu ISO (np. è wg ISO w ujęciu PN [4] dla oddania rosyjskiego э zmodyfikowano do postaci e', dogodnej dla komputera, podobnie ë przyjęło kształt e"). Nad-

Tabela transliteracji współczesnego alfabetu rosyjskiego na łaciński wg systemu ESEN

Lp.	Litery rosyjskie	Transliteracja	Przykłady
1	А	A	Атом = Atom
2	Б	B	Банк = Bank
3	В	V	Вход = Vchod
4	Г	G	Газ = Gaz
5	Д	D	Дом = Dom
6	Е (Ё)	E (Ë")	Ещё = Eszcze"
7	Ж	Zh	Железо = Zhelezo
8	З	Z	Завод = Zavod
9	И	I	Имя = Imja
10	Й	J	Каждый = Kazhdyj
11	К	K	Контур = Kontur
12	Л	L	Лавка = Lavka
13	М	M	Металл = Metall
14	Н	N	Наука = Nauka
15	О	O	Овощи = Ovoszczi
16	П	P	Патент = Patent
17	Р	R	Рига = Riga
18	С	S	Солнце = Solnce
19	Т	T	Так = Tak
20	У	U	Удар = Udar
21	Ф	F	Флакон = Flakon
22	Х	Ch	Хром = Chrom
23	Ц	C	Цвет = Cvet
24	Ч	Cz	Чёрный = Cze"rnyj
25	Ш	Sz	Шквал = Szkval
26	Щ	Szcz	Щётка = Szcze"tka
27	Ъ	"	Объезд = Ob"ezd
28	Ы	Y	Мыло = Mylo
29	Ь	'	Цель = Cel'
30	Э	E'	Этап = E'tap
31	Ю	Ju	Южный = Juzhnyj
32	Я	Ja	Ягуар = Jaguar
33	' (apostrof)	'	O'Кейси = O'Kejsi
34	' (twardy znak)	"	Об'ект = Ob"ekt

UWAGI DO TABELI

1. Lp. 34 — apostrof występuje niekiedy w piśmie maszynowym rosyjskim zamiast twardego znaku (lp. 27).
2. Drukarki serii ODRA 1300 i większości innych komputerów nie mają małych liter, będą więc wypisywać np. lp. 24 jako CZ, lp. 26 — SZCZ itd.
3. Zespoły liter łacińskich należy przedzielać znakiem =, jeśli wymagają tego względy jednoznaczności. Np. skrótowce (akronimy) rosyjskie п, ц, з i к, ш, ч transliteruje się dla drukarek standardowych ODRY jako PC=Z oraz KSZ=CZ, natomiast dla drukarek łańcuchowych Jednolitego Systemu wystarczy forma PCZ i KSzCz.
4. Zaleca się nie rozdzielać przy przenoszeniu do następnego wiersza zespołów liter łacińskich, odpowiadających jednej literze rosyjskiej.

5) Do normy tej opublikowano w 1973 r. zmianę nr 1, która zawiera m. in. postanowienie o dopuszczeniu na potrzeby epd odmiennej transliteracji trzech liter rosyjskich.

6) Centralny Instytut Dokumentacji Naukowo-Technicznej — poprzednik CIINTE i obecnych: CINTE oraz IINTE.

7) Dla dalekopisów — często używanych w systemach epd — należy stosować wersję ESEN-W z tą różnicą formalną, że w niektórych przypadkach (gdy dalekopis nie dysponuje wersalikami), wszystkie litery będą małe, a znak " zastąpiony zostanie dwukrotnie powtórzonym apostrofem (przewiduje to także Regulamin telegraficzny [8] w § 20 ust. 3 i 5).

mienić też należy, że nawet odrębne, nie występujące w ISO zestawy liter, jak np. „cz", „sz" mogą być z czasem przez ISO uznane za dopuszczalną mutację narodową systemu ISO (podobnie jak uznano specyficzne w Polsce „ch" dla rosyjskiego x oraz wariant anglo-amerykański, które umieszczone są w zaleceniu ISO [7]).

Należy jeszcze wyjaśnić sprawę kombinacji w systemie ESEN liter i związanych z nimi znaków diakrytycznych (po transliteracji): lp. 6 [E"] i lp. 30 [E']. Pozornie wydaje się, że te kombinowane litery mogą się mylić z transliteracją ъ i ь wg lp. 27 (") i lp. 29 ('). Otóż pomyłka jest tu wykluczona, gdyż w języku rosyjskim nigdy po literach Ё i Э nie występuje ani ъ (rosyjski „twardy znak") ani ь (rosyjski „miękki znak"), więc kombinacja E" i E' oznacza zawsze transliterację w systemie ESEN rosyjskich liter Ё i Э.

PERSPEKTYWY STOSOWANIA SYSTEMU ESEN

System transliteracji ESEN, dzięki dużemu podobieństwu do systemu transliteracji polskiej wg PN [4], może być stosunkowo łatwo wprowadzony w zastosowaniu do epd. Rozpowszechnienie w kraju może mu zapewnić opracowanie odpowiedniej normy branżowej (BN) lub państwowej (PN). W tym ostatnim przypadku, ze względu na omówione zalety oraz niewątpliwą uniwersalność mogłaby ona zastąpić wspomnianą już, obecnie obowiązującą normę PN.

Następnym krokiem mogłyby być starania o przyjęcie koncepcji ESEN na terenie międzynarodowym drogą zgłoszenia tego systemu do ISO i MCNTI poprzez polskich uczestników owych organizacji: PKNiM i CINTE. Starania takie mają pewne szanse powodzenia, gdyż zarówno w transliteracji ISO, jak i w transliteracji MCNTI są już obecnie stosowane lub dopuszczone odmienne wersje regionalne. Poza tym np. forma brzmieniowa „cz" (w ESEN odpowiednik transliteracyjny ros. ч) ma swoje tradycje w języku angielskim — tę formę zawierają angielskie wyrazy „Czechoslovakia" i „Czech".

Planować można także rozwój systemu ESEN przez objęcie transliteracją — w oparciu o jego zasady — dalszych języków posługujących się cyrylicą: białoruskiego, ukraińskiego i innych słowiańskich, a także mongolskiego i innych niesłowiańskich, o czym wspomniano już na wstępie niniejszego artykułu. Rozwijająca się współpraca międzynarodowa, m. in. w ramach RWPG, niewątpliwie przyspieszy pojawienie się takich potrzeb.

Odrębnym zagadnieniem, które może być rozwiązane pozytywnie w oparciu o zasady systemu ESEN, jest transliteracja na potrzeby komputerów 32-literowego alfabetu polskiego na 26-literowy alfabet łaciński. Sprawa jest dość pilna, gdyż stan obecny cechuje się niejednoznacznością tekstów: wydrukowany przez komputer wyraz LUK może oznaczać zarówno „luk" (okrętownictwo) jak i „łuk" (architektura, sport, geometria itp.) a skrótowiec SLZ może w tekście przedmaszynowym mieć kilkanaście różnych form (SLZ, SŁŻ, SLŻ, ŚLZ, ŚŁZ, ŚLŻ, ŚLŻ itd.). Odgadnięcie, o którą formę chodzi, okazuje się najczęściej niemożliwe prowadząc do omyłek i w konsekwencji do zmniejszenia skuteczności działania systemu epd, a co gorsze do istotnego podważenia zaufania użytkowników. Problem transliteracji alfabetu polskiego podjęto w kraju w latach 1973—1974 przy pracach nad zautomatyzowanym systemem informacyjno-wyszukiwawczym inte elektroenergetyki, nazywanym później ASIWELEN. Uzyskane rozwiązanie [12] było jednak bardzo dyskusyjne ze względu na zastosowanie znaków nieliterowych w funkcji liter w tekście przetransliterowanym. Autor niniejszego artykułu zamierza opracować, w oparciu o zasady systemu ESEN, system SEJEN obejmujący transliterację alfabetu polskiego na łaciński.

LITERATURA

[1] Gołąb Z., Heinz A., Polański K.: Słownik terminologii językoznawczej. Warszawa, PWN, 1968

[2] Starostin B. A.: Transkripcija sobstvennych imien, Wyd. 2. Moskva: Izdatelstwo „Kniga", 1965

[3] Diringer D.: Alfabet czyli klucz do dziejów ludzkości. Tłum. Hensel W., Warszawa, PIW 1972.

[4] Polska Norma PN-70/N-01201 Transliteracja słowiańskich alfabetów cyrylickich

[5] Dodatek nr 1. Zbiór znaków alfanumerycznych (s. 270). W.: Oprogramowanie mc ODRA serii 1300. COBOL. Opis języka. Publikacja nr 1301101. Wrocławskie Zakłady Elektroniczne ELWRO. Wyd. 2, marzec 1973

[6] Dodatek nr 2. Kod DKOI (s. 153). W: Drukarka wierszowa EC-7033. Dokumentacja techniczno-obsługowa. Tom 2. Opis drukarki wierszowej EC-7033. Zakłady Mechaniki Precyzyjnej MERA-BŁONIE, 1976 (?)

[7] Zalecenie ISO/R 9 — 1968 International system for transliteration of slavic cyrillic characters

[8] Ministerstwo Łączności. Dział IV telegraficzny. Tom 3. Służba ruchu telegramowego. Regulamin telegraficzny. Warszawa 1974, § 23 ust. 2

[9] Zalecenie NTP MCNTI 17—76 Pravila transliteracii bukv kirillovskogo alfavita bukvami latinskogo alfavita. Vestnik Meždunarodnogo Centra Naučnoj i Techničeskoj Informacii nr 2/1977, str. 29—32

[10] Instrukcja Centralnego Instytutu Dokumentacji Naukowo-Technicznej z dn. 01.03.1951 r. dot. transliteracji alfabetu rosyjskiego. W: Rytel Z. Opracowanie księgozbioru. Warszawa 1955 r. Stowarzyszenie Bibliotekarzy Polskich, str. 72—74

[11] Jodłowski S., Taszycki W.: Zasady pisowni polskiej i interpunkcji ze słownikiem ortograficznym. Wyd. 14. Wrocław, Ossolineum, 1961, str. 82—87

[12] Majewska I.: Zautomatyzowany system informacyjno-wyszukiwawczy elektroenergetyki — ASIWELEN. Aktualne Problemy Informacji i Dokumentacji nr 1/1975, str. 26, rys. 3.

MICHAŁ ZIĘBIŃSKI
Stołeczny Ośrodek Informacji i Techniki Obliczeniowej SOETO
Warszawa

PROGRES PR-5—system planowania i kontroli realizacji prac naukowo-badawczych

Współczesny rozwój krajów uprzemysłowionych opiera się w coraz większym stopniu na postępie technicznym. Postęp ten w większości dziedzin nauki i techniki zależy zaś od wyników prac dużych zespołów pracowników nauki, inżynierów i techników. W kompleksowych programach badawczych biorą często udział setki, a nawet tysiące specjalistów. Prawidłowa organizacja takich programów wymaga precyzyjnego planowania, koordynowania i kontrolowania realizacji setek, a nawet tysięcy tematów badawczych i rozwojowych.

Sterowanie takimi programami w oparciu o tradycyjne metody stosowane dotychczas w administracji prac naukowo-badawczych staje się bardzo uciążliwe i powoduje wiele trudności organizacyjnych. Konieczne jest więc zastosowanie nowoczesnych technik zarządzania w postaci systemów informatycznych.

W największych krajowych programach prac naukowo-badawczych, tzw. programach rządowych, biorą udział setki jednostek wykonawczych (instytutów, ośrodków badawczo-rozwojowych, wyższych uczelni, biur projektów i przedsiębiorstw) oraz tysiące naukowców i inżynierów.

Do koordynacji prac realizowanych w ramach tych programów zastosowano między innymi znaną od wielu lat metodę PERT i standardowe programy opracowane przez większość producentów komputerów. Standardowe systemy informatyczne PERT nie spełniają wielu wymogów stawianych przez jednostki odpowiedzialne za realizację programów rządowych, w konsekwencji więc zastosowanie metody PERT jest ograniczone. Dlatego też w celu usprawnienia zarządzania tymi programami opracowano specjalny system informatyczny PROGRES PR-5, służący do planowania i kontroli realizacji prac naukowo-badawczych. Sy-

stem PROGRES został zastosowany do sterowania rządowym programem PR-5 „Kompleksowy Rozwój Budownictwa Mieszkaniowego".

System uwzględnia wymogi związane z przyjętą strukturą organizacyjną realizacji „Programu", przepisy dotyczące planowania i rozliczania prac badawczych, jak również dotychczasowe doświadczenia w zakresie planowania, realizacji i rozliczania prac badawczych.

CEL I FUNKCJE SYSTEMU

Celem systemu PROGRES PR-5 jest usprawnienie całokształtu zagadnień związanych z organizacją zarządzania Programem PR-5. Podstawowe funkcje systemu są następujące:

— prowadzenie dokładnej, szczegółowej i aktualnej ewidencji wszystkich tematów, podtematów, zadań i etapów przewidzianych do realizacji w ramach programu
— ewidencja zleceń i umów dotyczących prac naukowo-badawczych
— szczegółowe planowanie zadań w okresach kwartalnych, rocznych i wieloletnich w różnych układach i przekrojach
— szczegółowa kontrola postępu realizacji prac w zakresie rzeczowym i finansowym
— szczegółowa kontrola załatwiania spraw formalnych (wysyłanie i potwierdzanie zleceń, zawieranie umów i aneksów, rozliczanie umów)
— kontrola terminów wdrożeń wyników prac badawczych
— prowadzenie wieloprzekrojowej sprawozdawczości z kolejnych etapów realizacji prac, jak również z aktualnego stanu formalnego przygotowania umów
— prowadzenie wieloprzekrojowej analizy szczegółowej i syntetycznej z realizacji programu.

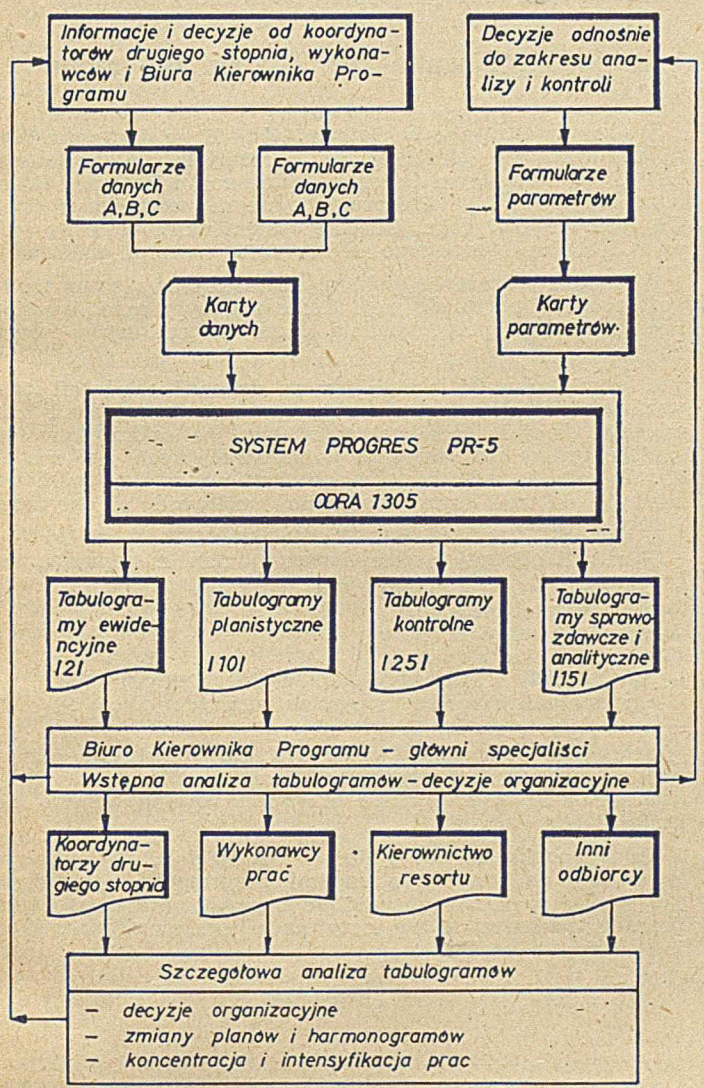
ZAKRES SYSTEMU

System obejmuje pełny cykl realizacji prac, począwszy od opracowania planu koordynacyjnego (pięcioletniego i rocznego), poprzez wysyłanie zamówień przez koordynatorów drugiego stopnia do wykonawców, potwierdzenie przyjęcia zamówienia, przygotowanie i podpisanie umowy. Następnie system obejmuje pełny cykl realizacji prac, rozliczenia finansowe i kontrolę wdrożeń wyników prac.

Systemem objęte są wszystkie prace badawcze realizowane w Programie PR-5 przez wszystkich wykonawców i podwykonawców.

System dostarcza tabulogramy wszystkim kontrahentom realizującym Program, m.in. takim, jak:
— biuro kierownika Programu
— koordynatorzy drugiego stopnia
— wykonawcy prac (wg ustaleń kierownictwa Programu)
— jednostki nadrzędne (wg ustaleń kierownictwa Programu).

Ogólny schemat działania systemu przedstawiono na rysunku poniżej.



Ogólny schemat działania systemu PROGRES PR-5

DANE WEJŚCIOWE DO SYSTEMU

W momencie wdrażania systemu dane wejściowe wprowadzane do systemu znajdowały się na wielu dokumentach, takich jak:
— pięcioletni plan koordynacyjny
— roczne plany koordynacyjne
— zlecenia i umowy generalne
— zlecenia i umowy o wykonanie tematów, podtematów i zadań
— zlecenia i umowy z podwykonawcami
— harmonogramy realizacji prac
— potwierdzenia przyjęcia zleceń
— umowy i aneksy do umów
— protokoły zdawczo-odbiorcze
— faktury
— harmonogramy i protokoły wdrożeń.

Dokumenty te nie są przystosowane do wymagań epd dlatego też dane z nich nanoszone są na trzy specjalne formularze, przystosowane do bezinstrukcyjnego przenoszenia na maszynowe nośniki informacji.

Po wdrożeniu systemu dane wprowadzane są z wyżej wymienionych dokumentów z pominięciem rocznych i pięcioletnich planów koordynacyjnych. Plany te są przygotowywane przez system na podstawie wcześniej wprowadzonych danych oraz w oparciu o uzupełnienia i korekty dokonywane bezpośrednio na tabulogramach i przenoszone na wspomniane formularze.

TABULOGRAMY WYNIKOWE

W wyniku przetwarzania system dostarcza 54 tabulogramy wynikowe w 7 podstawowych układach graficznych:

T-01 Podstawowy Rejestr Tematów

W układzie tym drukowanych jest 26 różnych tabulogramów, m.in.
T-07 Lista nie przygotowanych umów
T-09 Lista nie podpisanych umów
T-16 Lista umów zakończonych
T-17 Lista umów opóźnionych
T-23 Umowy planowane do zakończenia.

T-30 Szczegółowy Rejestr Tematów

W układzie tym drukowanych jest 18 różnych tabulogramów, z których najbardziej charakterystycznymi są:
T-33 Lista pozycji nie wykonanych w terminie
T-31 Lista wykonanych pozycji
T-40 Lista pozycji wdrożonych
T-44 Projekt planu sprzedaży
T-45 Plan sprzedaży.

Ponadto system dostarcza szereg tabulogramów zbiorczych obrazujących w sposób syntetyczny stan realizacji całego Programu:
T-50 Zestawienie wg grup kierunkowych
T-51 Zestawienie wg wykonawców
T-60 Analiza wg grup kierunkowych
T-61 Analiza wg wykonawców
T-70 Podtematy wykonane/wdrożone wg grup kierunkowych
T-71 Podtematy wykonane/wdrożone wg grup wykonawców.

Dla ułatwienia harmonogramowania i koordynacji prac opracowano dwa tabulogramy w formie harmonogramów graficznych:
T-80 Graficzny harmonogram realizacji podtematów
T-81 Graficzny harmonogram realizacji tematów.

Ponadto dla ułatwienia prac w zakresie uzyskiwania pozycji bibliograficznych dla poszczególnych tematów opracowano dwa tabulogramy dla służb informacji naukowo-technicznej Centralnego Ośrodka Informacji Budownictwa COIB (T-48 i T-49).

Z punktu widzenia podstawowych funkcji wszystkie tabulogramy systemu można podzielić na cztery grupy:

tabulogramy ewidencyjne	2
tabulogramy kontrolne	25
tabulogramy sprawozdawcze	17
tabulogramy planistyczne	10
Razem:	54

OGÓLNE ZASADY DZIAŁANIA SYSTEMU

System działa w ten sposób, że wszystkie dane dotyczące planowanych i realizowanych tematów, podtematów, zadań i etapów, znajdujące się na dokumentach źródłowych, przenoszone są przez pracowników koordynatorów drugiego stopnia na formularze danych. Formularze te przesyłane są do ośrodka obliczeniowego, gdzie dane przenoszone są na karty dziurkowane, a następnie przetwarzane. Liczba i zawartość tabulogramów wynikowych systemu sterowana jest za pomocą specjalnych parametrów, które umożliwiają wybranie określonego tabulogramu oraz potrzebnego układu cech klasyfikacyjnych i okresu czasu.

Po sprawdzeniu uzyskanych tabulogramów w ośrodku obliczeniowym przesyłane są one do biura kierownika Programu oraz do koordynatorów drugiego stopnia. Tabulogramy te stają się normalnymi materiałami roboczymi do celów ewidencyjnych, kontrolnych, planistycznych i sprawozdawczych.

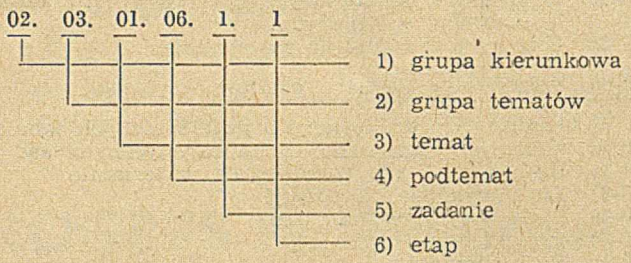
Wszystkie zmiany zachodzące w danych, takie jak:
— zmiany planów
— zawieranie umów
— korekty (aktualizowanie umów)
— rozliczanie umów itp.
są nanoszone na tabulogramy, a następnie po zakończeniu miesiąca przenoszone na formularze danych, które wysyłane są do ośrodka obliczeniowego celem wykonania kolejnych miesięcznych przeliczeń aktualizujących i uzupełniających poprzednie tabulogramy.

Powyższy schemat przepływu informacji widoczny jest na rysunku.

KLASYFIKACJA

Dla prawidłowego określenia każdego tematu, podtematu, zadania i etapu w systemie przyjęto następujące rodzaje klasyfikacji:
— numer tematu, podtematu, zadania i etapu
— numer umowy
— kody celu i kierunku prac
— kody przyczyn opóźnień
— cecha klasyfikująca.

Numer tematu, podtematu, zadania i etapu składa się z następujących sześciu członów:



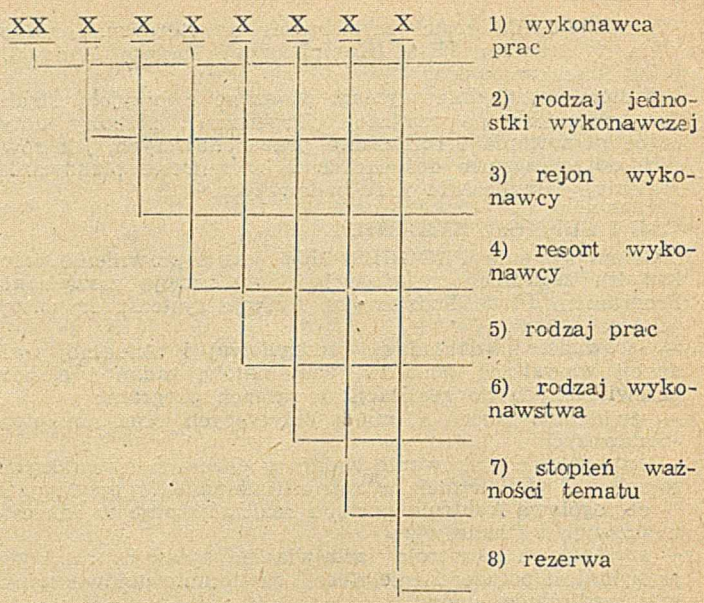
Numer umowy składa się zawsze z czterech znaków, np. 0251, 1235 itp. do 9998. Umowy zawierane są w zasadzie na podtemat, jednak system umożliwia zawieranie umów na temat lub zadanie.

Kody celu — stosowanych jest osiem kodów oznaczonych literami: K, P, I, W, U, R, N, T. Kody te służą do klasyfikacji podtematów lub tematów.

Kody kierunku prac — stosowane są cztery kody: S, P, M, R. Służą one do klasyfikacji podtematów i tematów.

Kody przyczyn opóźnień — stosowanych jest osiem kodów: Z, U, F, K, A, P, N, I.

Ponadto stosowana jest cecha klasyfikująca prac o następującej strukturze:



System pozwala na drukowanie tabulogramów typu T-01 i T-30 w układzie wg dowolnego elementu powyższego klasyfikatora lub dowolnej ich kombinacji. Umożliwia to otrzymywanie szeregu rejestrów oraz tabulogramów planistycznych, kontrolnych lub sprawozdawczych, np. w podziale na wykonawców, resorty, rejony wykonawców i rodzaje jednostek wykonawczych.

Dzięki wspomnianej już wcześniej możliwości agregowania danych dla dowolnych przedziałów czasowych system pozwala na bardzo dokładne planowanie prac, kontrolę ich realizacji oraz niezbędną sprawozdawczość.

OPRACOWANIE, WDROŻENIE I WYKORZYSTANIE SYSTEMU

System opracowano w latach 1976—77 w Warszawskim Przedsiębiorstwie Informatyki Przemysłu Budowlanego „ETOB" i Stołecznym Ośrodku Informacji i Techniki Obliczeniowej „SOETO". Wdrożenie systemu nastąpiło w drugiej połowie 1977 roku. Obecnie system eksploatowany jest w SOETO na komputerze ODRA 1305 w wersji taśmowej.

W trakcie wdrażania systemu natrafiono na szereg trudności natury organizacyjnej. Konieczne było m.in. ujednolicenie sposobu przygotowania planów, struktury i formy danych.

W chwili obecnej trudno jeszcze dokonać oceny efektów wynikających ze stosowania systemu, można jednak powiedzieć, że już teraz wpływ systemu na uporządkowanie szeregu spraw formalnych i administracyjnych jest znaczny. Ponadto zmniejszono pracochłonność szeregu sprawozdań i rozliczeń oraz stworzono możliwość syntetycznej i bardzo przejrzystej kontroli postępu prac.

W chwili obecnej system dostarcza m.in. następujące tabulogramy:
— planów pięcioletnich
— planów rocznych
— planów kwartalnych
— sprawozdań z realizacji planów kwartalnych
— sprawozdań z realizacji planów rocznych
— zestawień pozycji opóźnionych.

Ostateczne efekty stosowania systemu w zakresie planowania, kontroli koordynacyjnej i sprawozdawczości Programu PR-5 można będzie oczywiście ocenić dopiero po dłuższej eksploatacji.

Należy podkreślić, że system PROGRES PR-5 może być stosowany do sterowania innymi programami rządowymi czy resortowymi, jak również do usprawniania zarządzania pracami badawczymi w instytutach naukowo-badawczych lub wyższych uczelniach.

JAN WALUSZEWSKI
Politechnika Świętokrzyska
Kielce

Prawa twórców programów

Omówienie sytuacji prawnej twórców programów dla maszyn cyfrowych wymaga rozważenia ochrony prawnej tych programów. Zagadnienie to przedstawiłem bardziej szczegółowo w numerze 11/77 INFORMATYKI — tutaj przypomnę tylko najważniejsze kwestie.

OCHRONA PRAWNA PROGRAMÓW

Programy dla maszyn cyfrowych jako przedmioty intelektualne spełniają przesłanki zindywidualizowania, utrwalenia i posiadania wartości ekonomicznej, co pozwala uznać je za dobra prawne. Mogą być zatem chronione bezwzględnymi lub też względnymi prawami podmiotowymi.

W zakresie praw bezwzględnych ustawodawca — w zależności od tego, czy przedmiotem tych praw są dobra materialne czy niematerialne — stosuje odmienną technikę legislacyjną. W przypadku tych pierwszych istnieje ochrona generalna, gdyż wszystkie samoistne przedmioty materialne są przedmiotami praw rzeczowych, chyba że przepisem szczególnym zostały wyłączone z obrotu prawnego. Natomiast dobra niematerialne wolno uznać za przedmiot praw bezwzględnych wówczas, gdy istnieje norma prawna przyznająca taką ochronę pewnej kategorii tych dóbr.

W systemie prawa polskiego brak jest normy, która przyznawałaby ochronę bezwzględną programom dla maszyn cyfrowych. W szczególności art. 2 pkt. 5 ustawy o wynalazczości z dnia 19 października 1972 roku stanowi, że przepisów ustawy nie stosuje się do programów dla maszyn cyfrowych (Dz. U. 43/72, poz. 272). Nie ma natomiast tak wyraźnej normy w ustawie z dnia 10 lipca 1972 roku o prawie autorskim. Rozważmy zatem zagadnienie ochrony prawnoautorskiej programów (Dz. U. 34/52, poz. 234).

Uważam, że na tle przepisów ustawy brak jest podstaw uzasadniających pogląd, iż programy dla maszyn cyfrowych są w rozumieniu prawa utworami. Zgodnie z art. 1 przedmiotem prawa autorskiego jest „każdy utwór literacki, naukowy i artystyczny ustalony w jakiejkolwiek postaci". Przy wyodrębnianiu tych trzech kategorii utworów zastosowano różne kryteria: treści i związanej z nią struktury utworu przy utworach naukowych lub artystycznych oraz sposobu ustalenia przy utworach literackich. Ponieważ utwór literacki może być ze względu na treść albo utworem artystycznym albo naukowym, zbędne wydaje się wymienienie go w powołanym przepisie. Ograniczając zatem rozważania do dwóch kategorii dzieł: utworów artystycznych oraz utworów naukowych, zauważmy, iż spełniają one dwie główne funkcje: pierwsze — funkcję estetyczną, drugie zaś — poznawczą. Ponadto mogą spełniać jeszcze funkcję użytkową. Jednakże nie jest ona z punktu widzenia prawa autorskiego decydująca. Istotnym składnikiem utworów artystycznych i naukowych jest również pierwiastek inwencji twórczej, bez którego nie mogłyby one być uznane za przedmiot prawa autorskiego.

Programy dla maszyn cyfrowych nie spełniają podstawowych funkcji utworu będącego przedmiotem prawa autorskiego, ani nie wywołują wrażeń estetycznych u osób z nich korzystających ani też nie mają na celu odzwierciedlenia rzeczywistości i poznania praw nią rządzących. Ze względu na swą użyteczność zbliżają się natomiast do przedmiotów prawa wynalazczego. Również cecha oryginalności programów, jeśli niekiedy występuje, upodabnia je raczej do projektów wynalazczych, gdyż dotyczy idei rozwiązania, nie zaś formy, w jakiej zostało ono przedstawione. Gdyby zaś przyjmować ich ochronę prawnoautorską, analogiczną do przysługującej utworom naukowym, obejmowałaby ona przede wszystkim formę, nie sięgając do idei rozwiązania. Tylko bowiem w utworach artystycznych idea ta, czyli zgodnie z terminologią I. Kohlera — obraz zindywidualizowany, podlega ochronie prawnej. Wobec powyższego teza, że programy dla maszyn cyfrowych nie są utworami chronionymi prawem autorskim wydaje się uzasadniona i tym samym należy stwierdzić, iż w systemie prawa polskiego nie są one „chronionymi dobrami prawnymi".

Wydaje się natomiast, iż programy dla maszyn cyfrowych, ze względu na spełnianie przez nie wymaganych przesłanek, są „dobrami prawnymi" i mogą być przedmiotem praw podmiotowych względnych. Programy te wchodzą niekiedy w skład know-how, określanego przez B. Gawlika jako „wiedza techniczna o charakterze poufnym" (tajnym), bezpośrednio użyteczna w produkcji.

Powyżej starałem się przedstawić ochronę prawną, jakiej podlegają programy bez względu na osobę twórcy. Podmiotami praw, za pomocą których jest ona realizowana, są zawsze dysponenci programów. W dalszych rozważaniach zajmę się uprawnieniami przysługującymi wyłącznie tym osobom, które mają udział w tworzeniu programu dla maszyny cyfrowej.

SYTUACJA PRAWNA TWÓRCÓW PROGRAMÓW

Należy ustalić zakres nazwy *twórca programu dla maszyny cyfrowej*. Ponieważ rozważamy programy jako twory intelektualne, zakresem nazwy *twórca programu dla maszyny cyfrowej* objęte zostaną jedynie te osoby, których wkład w powstawanie programu ma charakter twórczości intelektualnej. Co to jest *twórczość intelektualna*? Niestety mimo iż jest to termin języka prawnego, występujący np. w ustawie o wynalazczości, brak jest dotychczas znaczenia powszechnie akceptowanego.

W języku potocznym z pojęciem *twórczości* spotykamy się przy wszelkiego rodzaju pracy — tak umysłowej, jak i fizycznej — jeżeli powstają pewne nowe, poprzednio nie istniejące przedmioty. T. Kotarbiński pisze, że: „działał twórczo, kto czynem osiągnął coś nowego, a cennego m. in. tym właśnie, że nowego". Dostrzega jednak, że natężenie cechy *twórczości* może być różne, a przyznając, iż nawet przepisanie tekstu na maszynie nosi w sobie szczyptę nowości, dodaje: „jakaż odległość jednak dzieli tę szczyptę nowości od rękopisu, w którym badacz wypowiedział po raz pierwszy doniosłe odkrycie".

Na podstawie wypowiedzi T. Kotarbińskiego widać, że tego terminu można używać zarówno w celu określenia pewnej działalności ludzkiej, jak i wytworu tej działalności. A. Kopff uważa, że „przez słowo *twórczość* zwykło

się dość często rozumieć proces twórczy, czyli zespół czynności psychofizycznych, których przedmiotem jest twórca". W. Tatarkiewicz wskazuje, iż dopiero w czasach nowożytnych, a właściwie od XIX wieku, poczęto posługiwać się tym terminem dla określenia działalności ludzkiej w dziedzinie sztuki i to wyłącznie w tej dziedzinie. Dopiero w naszym stuleciu zakresem terminu *twórczość* objęto „wszelkiego rodzaju ludzkie czynności i wytwory nie tylko artystów, ale także uczonych i techników.

T. Tatarkiewicz, tak jak i inni — podkreśla, że nowość ma dla pojęcia *twórczości* znaczenie konstytutywne. Stwierdza jednak zarazem, że o ile każda twórczość implikuje nowość, to nie każda nowość implikuje twórczość. Nowość podlega stopniowaniu i można powiedzieć, że „*twórczość* jest wysokim stopniem nowości". Drugą cechą konstytutywną *twórczości* jest zdaniem Tatarkiewicza energia umysłowa zużyta na wytworzenie nowej rzeczy. Energia ta, tak jak i nowość, stanowi miarę *twórczości*. „Obie zaś — nie dają się mierzyć, oceniane mogą być jedynie intuicyjnie".

Przedstawione powyżej, zawarte w potocznym języku lub rozprawach filozoficznych sposoby rozumienia terminu *twórczość*, nie są wystarczające dla prawnika, który chce posługiwać się nim jako terminem technicznoprawnym. Jednakże brak jest w literaturze prawniczej precyzyjnego określenia. Przedstawione na przykład przez A. Chobota ujęcie tego terminu ograniczone jest do dziedziny twórczości wynalazczej i polega na zestawieniu katalogu cech, które muszą charakteryzować pewną pracę, aby można ją było uznać za *twórczość wynalazczą*. Podkreśla on, że ustalenie wielkości natężenia tych cech — takiej, która rozstrzyga o twórczości — nastręcza poważne trudności.

Inaczej pojmuje się pojęcie *twórczości* w literaturze prawa autorskiego. Przez *twórczość* rozumie się tu nadanie utworowi piętna osobistego jego twórcy. Głębsza analiza tego stanowiska prowadzi jednak do wniosku, że przy takim ujęciu terminu podstawowe znaczenie przy określaniu jego desygnatów ma stopień, w jakim osobiste piętno twórcy przejawia się w utworze.

Sądzę, że w związku ze stosowaniem prawa najbardziej przydatne będzie posługiwanie się terminem *twórczość* jako pojęciem typologicznym. Cechami przysługującymi pojęciu *twórczość* byłyby nowość, oryginalność, styl charakterystyczny dla twórcy, pozwalający na jego nieomylne zidentyfikowanie i inne. Można również wyobrazić sobie biegunowo przeciwne pojęcie typologiczne *nietwórczość*, którego idealnemu typowi nie przysługiwałyby żadne z wymienionych cech nawet w minimalnym stopniu. Na skali ograniczonej owymi przeciwstawnymi typami idealnymi mieściłyby się wszystkie desygnaty pojęcia „praca". Aby ustalić, czy dana praca jest pracą twórczą, badaliśmy czy między występującymi w niej cechami a cechami przysługującymi pojęciu *twórczość* zachodzi większa adekwatność niż pomiędzy jej cechami a cechami pojęcia *nietwórczość*.

W konkluzji można stwierdzić, że jakkolwiek w każdej efektywnej pracy tkwią elementy *twórczości*, to jednak nie zawsze występują one w takim natężeniu, aby uzasadniały użycie w stosunku do niej tego terminu. Z punktu widzenia ochrony prawnej twórcy istotne znaczenie ma natomiast tylko taka działalność ludzka, którą określamy jako twórczość intelektualną. Chociaż, bowiem, dana efektywna praca intelektualna zawiera w sobie pierwiastki twórcze, podmiotowi, który ją wykonywał, nie będzie — w myśl zasady de minimis non curat practor — przysługiwała ochrona prawna, jeśli pierwiastki te będą występowały w znikomej ilości.

Istnieje kwestia miary, za pomocą której będziemy ustalać, czy daną działalność można uważać za pracę twórczą. Można ją zdefiniować jedynie dla poszczególnych rodzajów twórczości intelektualnej, nie da się natomiast sformułować ogólnie. W przypadku na przykład działalności zmierzającej do zbudowania programu o tym, czy konkretny wkład pracy danej osoby ma charakter twórczy, decydować mogą jedynie specjaliści-informatycy. Wbrew pozorom dokonanie przez nich takiej oceny nie jest szczególnie trudne, gdyż na podstawie literatury i praktyki mogą oni dość precyzyjnie ustalić, czy wykonana praca ma cechy właściwe pojęciu *twórczości* i to w stopniu uzasadniającym uznanie jej za twórczą. Tego rodzaju metoda określania twórczego charakteru pracy funkcjonuje od wielu lat w praktyce biur projektowych przy ocenie projektów architektonicznych.

Osobami, w stosunku do których można używać nazwy *twórcy programów dla maszyn cyfrowych*, są: analityk-projektant, będący głównym projektantem systemu, projektanci-programiści oraz programiści sensu stricto.

Analityk-projektant, często ekonomista z wykształcenia, musi posiadać gruntowną wiedzę z zakresu organizacji przedsiębiorstw i tradycyjnych oraz elektronicznych sposobów przetwarzania danych. Po przeanalizowaniu struktury wybranego przedsiębiorstwa sporządza on schemat jego modernizacji, ponieważ jedynie wyjątkowo będzie można zastosować elektroniczne przetwarzanie danych w istniejącym systemie organizacyjnym. Analiza ta, jak również opracowanie założeń systemu przetwarzania danych, może niewątpliwie zawierać pierwiastki twórcze. Dodam, że niektóre z zastosowanych przez niego rozwiązań mogą być przedmiotowo nowe, jednak wobec wyraźnego brzmienia przepisu art. 2 pkt. 4 ustawy o wynalazczości nie podlegają one przepisom tej ustawy, gdyż są usprawnieniami z dziedziny organizacji administracji.

Projektanci-programiści, zazwyczaj również z wykształceniem ekonomicznym lub matematycznym, przygotowują dokumentację techniczną, w której muszą dokładnie sprecyzować ogólne założenia opracowane w pierwszym etapie budowy systemu. Praca ich może być także twórcza. Rozwiązują oni bowiem takie problemy, jak np. przedstawienie poszczególnych elementów schematu organizacyjnego przedsiębiorstwa oraz przebiegów informacji w systemie. Jest to więc, stosując odległą analogię do terminologii kohlerowskiej, opracowanie treści w nowej formie.

Także praca programistów, którzy są najczęściej matematykami i rozpisują opracowane przez projektantów fragmenty systemu na język maszynowy, przedstawiając poszczególne schematy blokowe w postaci ciągów rozkazów dla maszyny, może być niekiedy uznana za twórczą. Mogą oni dokonywać nowych, indywidualnych rozwiązań, przedstawiając na przykład przebiegi informacji w krótszy lub prostszy sposób. Inwencję swą mogą zwłaszcza wykazać przy organizowaniu współpracy programów, aczkolwiek przeważnie wykorzystują do tych celów standardowe systemy operacyjne.

Praca wykonywana przez te trzy kategorie twórców zakłada w nich wysokie kwalifikacje zawodowe, wiedzę i doświadczenie. Oczywiście im kwalifikacje te są wyższe, tym wyniki pracy są lepsze i tym większe prawodopodobieństwo, iż rozwiązania przez nich przygotowane, charakteryzujące się większą użytecznością będą nosiły znamię nowości przedmiotowej.

Jeżeli dokonane rozwiązanie charakteryzuje się nowością przedmiotową, czyli jest sposobem postępowania po raz pierwszy w świecie zastosowanym, nie ma wątpliwości, iż działalność, której wynikiem jest to rozwiązanie, była pracą twórczą. Nie oznacza to jednak, by warunkiem sine qua non uznania jakiejś pracy za twórczą było osiągnięcie w jej wyniku rozwiązań nowych przedmiotowo.

Badając bowiem, czy pewna działalność jest pracą twórczą, kwestię nowości rozwiązania powinniśmy rozważać z punktu widzenia podmiotu wykonującego tę pracę. *Twórczość* jest wykluczona jedynie wtedy, gdy autor rozwiązania posłuży się znanym identycznym sposobem postępowania.

Zagadnienie znajomości poprzednich rozwiązań można rozstrzygać przyjmując domniemanie faktyczne, iż specjaliście w danej dziedzinie znane są podstawowe publikacje w tym zakresie.

W wywodach zawartych w niniejszej części pracy świadomie skoncentrowałem uwagę na twórczości traktowanej jako działalność ludzka czyli pewien proces. *Twórczość* można także ujmować jako wytwór działalności ludzkiej. W takim wypadku są to rezultaty twórczej pracy autora.

Przy określaniu, które rezultaty działalności człowieka można nazwać *twórczością* musimy odwołać się do charakteru procesów, które doprowadziły do powstania tych rezultatów. Tylko bowiem efekty samodzielnej, oryginalnej i posiadającej osobiste piętno pracy, w znaczeniu czynnościowym, uznamy za twórczość w sensie przedmiotowym. Przy szukaniu podstaw ochrony interesów twórców programów dla maszyn cyfrowych nie na wiele zda się jednak badanie właściwości rezultatów ich pracy, czyli programów dla maszyn cyfrowych, skoro nie są one „chronionymi dobrami prawnymi", ponieważ nie są objęte przepisami prawa wynalazczego i autorskiego.

PRAWA OSOBISTE

W artykule 23 Kodeksu Cywilnego zawarta jest formuła: „dobra osobiste człowieka, jak w szczególności (...) twórczość naukowa, artystyczna, wynalazcza i racjonalizatorska, pozostają pod ochroną prawa cywilnego niezależnie od ochrony przewidzianej w innych przepisach". W doktrynie prawa cywilnego nie ma wątpliwości co do tego, że przepis ten wymienia dobra osobiste jedynie przykładowo i że oprócz wyliczonych w nim istnieją jeszcze inne dobra prawnie chronione. Tak więc zasadne jest twierdzenie, że oprócz wyżej wymienionych rodzajów twórczości intelektualnej, także i inne jej odmiany stanowią dobra osobiste podlegające ochronie z art. 23 k.c. Wydaje się również, że ochrona prawna przysługuje twórcom niezależnie od tego, czy rezultatem ich twórczości są przedmioty niematerialne chronione prawami bezwzględnymi (w szczególności — utwory w rozumieniu prawa autorskiego) czy projekty wynalazcze w myśl przepisów ustawy o wynalazczości.

Artykuły 23 i 24 k.c. chronią dobro osobiste w postaci stosunku twórcy do pewnego przedmiotu, nie zaś ten przedmiot. Twórca może mieć osobisty stosunek zarówno do swych dzieł, jak i do samego procesu twórczego. W jednym i drugim przypadku podlega on ochronie. Proces twórczy, czyli twórczość, istnieje w odniesieniu do jakichś tworzonych przedmiotów. Przedmioty te, nawet gdy same nie są chronione, stanowią punkt odniesienia dla praw osobistych twórcy.

W przypadku programów dla maszyn cyfrowych można stwierdzić, że chociaż nie są one chronionymi dobrami prawnymi, stanowią pośredni punkt odniesienia dla bezwzględnych i wyłącznych praw osobistych służących twórcy. Osobisty stosunek twórców do wykonywanych przez nich programów, jak i do samego procesu ich budowy, jest zatem, moim zdaniem, chroniony przepisami art. 23 i 24 k.c. — pod warunkiem, że podczas przygotowania programów przejawiły się pierwiastki twórcze w stopniu uzasadniającym uznanie tej pracy za twórczą.

Dla określenia treści praw osobistych służących twórcom programów dla maszyn cyfrowych posłużyć się można konstrukcją analogiczną do występującej w prawie autorskim w odniesieniu do „dobra osobistego". Tego rodzaju analogia iuris zdaje się być o tyle uzasadniona, że konstrukcje prawnoautorskie są najbardziej odpowiednie dla opisu sytuacji prawnej twórców programów dla maszyn cyfrowych.

Nie zachodzi potrzeba wyliczania uprawnień przysługujących twórcom programów, lecz wystarczy ogólne stwierdzenie, iż prawo to ma strzec przed wszelkim „zamącaniem stanu psychicznego" czy „naruszeniem życia uczuciowego". Oczywiste natomiast jest, iż na twórcy ciąży dowód wykazania, że działanie sprawcy rzeczywiście narusza jego dobro osobiste, a okoliczność ta będzie podlegała wszechstronnej ocenie sądu.

Zaproponowana forma ochrony dóbr osobistych twórców programów dla maszyn cyfrowych skłania do zastanowienia się nad tym, czy taka ochrona nie wywrze niepożądanego wpływu na sytuację prawną podmiotów, na rzecz których powstają prawa majątkowe. W szczególności idzie o to, czy korzystanie przez twórców z uprawnień osobistych nie uniemożliwi korzystania z programów osobom uprawnionym.

Wśród uprawnień składających się na ogólne autorskie prawo osobiste, służące do ochrony szeroko ujętego osobistego stosunku autora do jego twórczości, jako najważniejsze wymienia się często droit de la paternité, droit de l'integrité oraz droit de rester inédir. O ile w przypadku pierwszego z nich wykonywanie go, czyli domaganie się uznawania, iż dany przedmiot (w naszym przypadku — program) jest dziełem określonej osoby, z reguły nie powinno prowadzić do wyżej wspomnianego konfliktu, to co do pozostałych uprawnień powstawania takiego konfliktu jest możliwe. Prawo do integralności utworu stanowiłoby bowiem podstawę do domagania się przez twórcę, aby opracowany przez niego program nie był w dalszym toku prac nad systemem zmieniany ani uzupełniany. Podobnie droit de rester inédir uzasadniałoby ewentualny sprzeciw autora programu do jego „publikacji", a tym samym całkowicie uniemożliwiałoby stosowanie programu, skoro „publikacją" byłoby także jego ujawnienie między pracownikami przedsiębiorstwa, w którym program jest opracowywany. Wydaje się, iż rozstrzygając powyższą kwestię, należy wyjść od analizy treści stosunku łączącego twórcę programu dla maszyny cyfrowej z podmiotem, któremu służą prawa majątkowe względem tego programu.

Ponieważ w praktyce nie występują twórcy programów pracujący na własny rachunek, regułą jest, iż zatrudnieni są oni w uspołecznionych (w krajach kapitalistycznych także prywatnych) przedsiębiorstwach na podstawie umowy o pracę. Możliwe są przypadki zawierania z twórcami programów umów o dzieło, natomiast mało prawdopodobne jest zawieranie umów-zleceń. Charakter wykonywanych świadczeń sprawia bowiem, iż na podstawie zawieranych z nimi umów powstają zobowiązania rezultatu.

Uważam, iż można zasadnie reprezentować pogląd, że z istoty umowy o pracę wynika powstanie dla pracodawcy prawa własności rzeczy stanowiących rezultat pracy pracownika. Przy umowie o dzieło zamawiający nabywa własność dzieła wskutek wykonania go przez wykonującego. Jeśli rezultatem pracy są przedmioty niematerialne, na rzecz wymienionych podmiotów powstaje prawo do swobodnego dysponowania tymi przedmiotami. Prawo to ma charakter względny i jest skuteczne jedynie w stosunku do twórcy, w stosunku do innych osób interesy dysponenta chronione są jedynie przepisami o odpowiedzialności ex delicto. Przyjęcie odmiennego stanowiska pociągnęłoby za sobą uznanie, iż pracodawca lub zamawiający w celu uzyskania prawa do rezultatów pracy dłużnika powinien zawrzeć z nim dodatkową umowę o przeniesienie praw.

W oparciu o przedstawiony pogląd można, moim zdaniem, przyjąć, że w treści umowy o pracę, mimo braku w tym zakresie wyraźnych oświadczeń woli stron, zawarte jest zezwolenie twórcy na dokonywanie w wytworzonym przez niego przedmiocie wszelkich zmian, które pracodawca uzna za wskazane i konieczne dla należytego wykorzystania tego przedmiotu. W umowie mieści się także zgoda twórcy na to, że po otrzymaniu tego przedmiotu pracodawca będzie wyłącznie decydował o jego ujawnieniu i stosowaniu. Identyczna sytuacja występuje przy umowie o dzieło.

W tym miejscu należy wyraźnie podkreślić, że tego rodzaju uprawnienia powstają po stronie pracodawcy lub zamawiającego dzieło jedynie wtedy, gdy dany przedmiot nie jest chroniony za pomocą bezwzględnego prawa podmiotowego, takiego np. jak określone w prawie autorskim lub wynalazczym. Rozważam bowiem sytuację, która istnieje wówczas, gdy twórcy służą wyłącznie prawa osobiste z art. 23 i 24 k.c.

Przesłankami roszczeń z art. 24 § 1 k.c. są zagrożenie lub naruszenie prawa osobistego oraz bezprawność działania. Rozważmy przypadek gdy w programie dokonane zostały zmiany, które obniżają wartość zastosowanych w nim rozwiązań, co w oczywisty sposób narusza autorskie dobro osobiste twórcy. Czy służą mu roszczenia z art. 24 § 1 k.c.? Trzeba usalić, czy działanie pracodawcy, który zlecił dokonanie zmian w programie, było bezprawne. Sądzę, iż zgodnie z przedstawionym wyżej stanowiskiem, należy uznać, że działanie pracodawcy lub zamawiającego dzieło nie jest w tej sytuacji bezprawne, ponieważ wykonuje on swoje prawo podmiotowe. Prawem przez niego wykonywanym jest prawo dysponowania programem, w ramach którego służą mu uprawnienia do dokonywania zmian i decydowania o „publikacji".

Na zakończenie wywodów dotyczących zagadnienia wpływu ochrony osobistej twórców programów na sytuację prawną podmiotów posiadających w stosunku do programów prawa majątkowe należy nadmienić, że przedstawione w tej kwestii stanowisko w niczym nie ogranicza autorskich praw osobistych służących twórcom w stosunku do innych osób. W szczególności twórcy mogą podnosić wynikające z droit de la paternité, droit de l'integrité lub droit de la rester inédit roszczenia przeciwko osobom, które naruszają ich uprawnienia nawet jeśli dysponent programu nie będzie występował przeciw tego rodzaju działaniom.

W związku z omawianiem praw osobistych twórców powiedziałem, że prawa majątkowe dotyczące programów dla maszyn cyfrowych przysługują z reguły innym podmiotom niż twórcy. Wynika to z dwóch przyczyn. Po pierwsze, nie ma normy prawnej przyznającej bezwzględne i wyłączne prawo majątkowe na programach dla maszyn cyfrowych ich twórcom ani też żadnym innym podmiotom. Jak wspomniałem, tego rodzaju norma byłaby konieczna w celu uznania programów za „chronione dobra prawne". Po drugie, z umowy o pracę lub z umowy o dzieło wynika, że druga strona umowy uzyskuje prawo własności lub — w przypadku programów traktowanych jako przedmioty intelektualne — prawo swobodnego dysponowania wytworem twórcy. Tylko wyjątkowo, co w praktyce nie zdarza się, twórca programu tworzący go samodzielnie mógłby swobodnie dysponować programem, w szczególności utrzymywać go w tajemnicy i czerpać korzyści majątkowe na podstawie zawieranych umów.

W przeważającej liczbie przypadków twórcy programów przysługuje prawo jedynie do wynagrodzenia za świadczenia spełnione w wykonaniu umowy. W umowie o dzieło wysokość wynagrodzenia, ustalona w wyniku porozumienia stron, może być uzależniona od jakości programu. Jest to rozwiązanie najbardziej godne aprobaty, choć nie wolno zapominać, że brak jest obiektywnych kryteriów pozwalających jednoznacznie ocenić jakość programu. Wydaje się jednak, że okoliczność ta nie stanowi nieprzezwyciężonej przeszkody, ponieważ zawsze można się odwołać do opinii ekspertów, którzy zbadają oryginalność, użyteczność i bezbłędność programu. Jak już w innym miejscu wspomniano, podobne postępowanie stosowane jest np. w praktyce biur projektowych przy kwalifikowaniu projektów architektonicznych.

W Polsce ogromna większość programów powstaje w przedsiębiorstwach państwowych, gdzie wykonują je twórcy zatrudnieni na podstawie umowy o pracę. Wysokość wynagrodzenia, jakie im przysługuje, uzależnione jest od stanowiska, wykształcenia i stażu pracy w zawodzie. Taki system płac, stosowany powszechnie w gospodarce uspołecznionej, w tym przypadku jest szczególnie nieskuteczny. Nie stwarza on bowiem zinstytucjonalizowanych środków wpływania na jakość pracy twórców programów. Przy stosunkowo dużym udziale inwencji twórczej występującej przy budowie programów osiągnięte dobre efekty pracy nie pociągają za sobą wyraźnych skutków w dziedzinie wynagrodzeń. Co prawda doświadczony kierownik jednostki organizacyjnej wykonującej programy potrafi ocenić jakość pracy poszczególnych twórców programów, lecz w ramach przysługujących mu uprawnień ma niewielkie możliwości przyznania wyróżniającemu się pracownikowi wyższego wynagrodzenia.

* * *

Na zakończenie pragnę zrekapitulować omawiane zagadnienia i przypomnieć główne tezy.

- Programy dla maszyn cyfrowych jako przedmioty intelektualne są dobrami prawnymi ze względu na to, że spełniają przesłanki zindywidualizowania, utrwalania i posiadania wartości ekonomicznej. Oznacza to, iż mogą być one przedmiotem praw podmiotowych względnych.
- Odnośnie do ochrony prawnej programów, uważam, że nie są one chronione wyłącznymi prawami bezwzględnymi, w szczególności nie stosuje się do nich przepisów prawa wynalazczego i autorskiego. Przysługuje im natomiast ochrona wynikająca z przepisów o odpowiedzialności deliktowej, zawartych w k.c. i ustawie o zwalczaniu nieuczciwej konkurencji.
- Za twórców programów dla maszyn cyfrowych uważać należy analityków, projektantów i programistów, pod warunkiem, iż ich udział w budowie programów nosi charakter twórczości intelektualnej.
- Pierwiastek twórczości intelektualnej, występujący w pracy twórców programów, uzasadnia przyznanie im ochrony ze względu na ich dobra osobiste polegające na osobistym stosunku do dzieła, tj. programu oraz do procesu twórczego. Ochronę tę normują art. 23 i 24 k.c.
- Twórcom programów dla maszyn cyfrowych nie przysługują prawa majątkowe w stosunku do programów, lecz mają oni jedynie prawo do wynagrodzenia ustalonego w umowie łączącej twórcę z przedsiębiorstwem, w którym program jest wykonywany.

Współdziałanie ze SPISEM

Komisja Informatyki przy Zarządzie Głównym PTE i Ośrodek Badawczo-Rozwojowy Systemu Państwowej Informacji Statystycznej (SPIS) organizuje w dniach 24—26 października 1978 r. ogólnopolskie seminarium naukowe na temat:

„Rządowy System Informatyczny SPIS a systemy informatyczne resortów (problemy spójności i współdziałania rządowego systemu informacyjnego SPIS z resortowymi systemami informatycznymi)".

Tematyka obrad będzie obejmować problemy spójności informacyjnej, językowej, metodycznej, funkcjonalnej, organizacyjnej, technicznej i metody projektowania. Wymienione problemy to jednocześnie warunki, które muszą być spełnione, aby SPIS mógł współdziałać z systemami resortowymi. Oto pokrótce ich charakterystyka:

- **spójność informacyjna**, a więc zgodność informacji gromadzonych w SPIS z potrzebami informacyjnymi odbiorów informacji, reprezentowanych przez systemy resortowe, i odwrotnie — zgodność informacji gromadzonych przez systemy resortowe z potrzebami innych użytkowników, zaspokajanymi przez statystykę państwową; zgodność ta dotyczy zakresu informacji, szczegółowości, układów grupowania itd;
- **spójność językowa** polega na stosowaniu w systemach współdziałających jednolitych języków, na przykład katalogów informacji, klasyfikacji i nomenklatur, rejestrów jednostek gospodarczych;
- **spójność metodyczna** polega na zgodności metod opisu obiektów lub zjawisk, o których dane zgromadzone są w systemach informatycznych, metod kontroli danych, algorytmów sprowadzania danych do porównywalności, szacowania i przeliczania informacji itd.; spójność metodyczna warunkuje utrzymanie merytorycznej porównywalności danych między systemami;
- Dzięki **spójności funkcjonalnej** powinno uzyskać się ustalenie trybu funkcjonowania systemów współdziałających, uwzględniającego wzajemne uwarunkowania w gromadzeniu i przetwarzaniu informacji, między innymi niesprzeczne harmonogramy gromadzenia i przekazywania danych, stosowanie obiegu informacji, umożliwiającego jak najlepszą kontrolę kompletności i wiarygodności danych, zaspokajanie potrzeb informacyjnych ogniw pośrednich w procesie gromadzenia i przekazywania informacji na szczebel centralny (o ile jest to technologicznie uzasadnione);
- **spójność organizacyjna** polega nie na dążeniu do podobnych struktur organizacyjnych resortowych systemów informatycznych, lecz na zgodności rozwiązań organizacyjnych z potrzebami funkcjonowania systemu informatycznego; spójność organizacyjna jest więc instrumentem do uzyskania spójności funkcjonalnej;
- **spójność techniczna** umożliwia techniczną realizację przepływów informacji między systemami (chodzi tu o jednolite standardy techniczne, zwłaszcza dotyczące maszynowych nośników i teletransmisji);
- **spójność metod projektowania** polega na zapewnieniu jednolitych standardów opracowania dokumentacji, opisu zbiorów danych, identyfikacji danych oraz „wymusza" na projektantach systemów informatycznych korzystanie z jednolitej bazy normatywnej.

Podczas obrad plenarnych zostaną wygłoszone referaty przygotowane przez projektantów SPIS i przedstawicieli ośrodków wiodących w zakresie projektowania i wdrażania systemów informatycznych. Zaprezentowane zostaną koncepcje i doświadczenia uzyskane w resortach o różnej specyfice i modelu funkcjonowania systemów informacyjnych, o różnym modelu zarządzania i koordynacji branżowej.

Oprócz obrad plenarnych przewidziane są sesje tematyczne:

- Problemy przepływu informacji między SPIS i systemami informatycznymi resortów
- Wspólne systemy instrumentalne jako narzędzie koordynacji i spójności SPIS i resortowych systemów informatycznych
- Problemy budowy i eksploatacji wspólnej bazy normatywnej (klasyfikacje, nomenklatury, rejestry). Unifikacja ewidencji podstawowej
- Organizacyjne i techniczne problemy przepływów informacji.

Przewiduje się także zorganizowanie dyskusji panelowej na temat „Granice spójności wielkich systemów informatycznych".

ANDRZEJ KUKUŁA
Zakład Problemów Organizacji i Zarządzania PAN
Bytom

Język SPECOL narzędziem użytkownika

Ostatnio oprócz bogatego oprogramowania specjalistycznego, stanowiącego narzędzie projektantów i programistów systemów informatycznych, coraz większe znaczenie ma oprogramowanie przeznaczone dla użytkownika, który nie jest informatykiem — profesjonalistą. Z uwagi na przeznaczenie tego oprogramowania jego główną cechą jest to, iż jest łatwe w zastosowaniu. Jednym z narzędzi, które pozwalają szerokiemu gronu ludzi korzystać z komputera, a zwłaszcza z informacji przechowywanych w jego pamięci zewnętrznej, jest język SPECOL (skrót od *SPEcial Customer Oriented Language).*

Oprócz funkcji przeglądania i wyszukiwania informacji SPECOL umożliwia również realizację pewnych dalszych operacji na danych, a więc samodzielne zaprogramowanie przez użytkownika określonego zakresu przetwarzania. Przetwarzanie to ma co prawda ograniczony zasięg, niemniej pozwala uzyskiwać bardzo szybko odpowiedź na pytania, a zwłaszcza wyeliminować pośrednictwo specjalisty-informatyka. Istotne jest również to, że SPECOL pozwalając użytkownikowi włączyć się bezpośrednio do prac programowych zwiększa jego zaufanie do informatyki w ogóle.

Z doświadczenia w pracy z pakietem SPECOL wynika, że może on być również wykorzystywany przez zawodowych programistów. Jest on przydatny zwłaszcza w testowaniu kartotek, zarówno w fazie ich tworzenia, jak i w trakcie eksploatacji.

ŻĄDANIA W JĘZYKU SPECOL

Przed opracowaniem programu (żądania) w języku SPECOL użytkownik powinien zapoznać się z zawartością informacyjną kartoteki oraz sposobem zorganizowania w niej danych. Następnie w celu uzyskania interesującej go informacji musi napisać żądanie specolowe, tzw. zestaw instrukcji akceptowanych przez kompilator SPECOL.

Załóżmy, że w pewnej instytucji istnieje kartoteka osobowa zawierająca informacje o pracownikach, a wśród nich takie, jak: nazwisko, imię, miejsce zamieszkania, zarobek, stanowisko pracy. Użytkownik chce wiedzieć ilu mężczyzn mieszkających w Katowicach zarabia więcej niż 4000 zł. To pytanie z punktu widzenia SPECOLU będzie miało następującą postać: „Wyszukać i policzyć ile jest w kartotece zapisów o pracownikach płci męskiej, mieszkających w Katowicach, o zarobku większym niż 4000 zł."

Precyzowanie żądania nie odbiega w sposób istotny od przeglądania normalnej ręcznie obsługiwanej kartoteki.

Rozróżniane są trzy typy zapytań: przeszukiwanie, zliczanie i sumowanie. Przeszukiwanie zbioru powoduje odnalezienie i wydrukowanie żądanych informacji jeżeli tylko istnieją w kartotece. Zliczanie powoduje wyszukanie określonych danych z kartoteki oraz podanie liczby ich wystąpień; na przykład policzyć ilu jest w przedsiębiorstwie pracowników urodzonych w 1912 roku. Sumowanie umożliwia obliczenie sumy wartości określonych danych; na przykład: zsumować zarobki wszystkich pracowników urodzonych w roku 1945.

Żądanie w języku SPECOL może być proste, to znaczy może zawierać pytanie tylko jednego typu lub też składać się z kombinacji przeszukiwania i zliczania bądź przeszukiwania i sumowania.

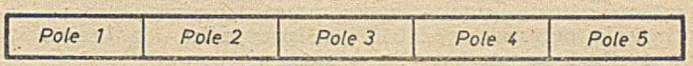
ZBIORY W JĘZYKU SPECOL

Kartoteki nie muszą być specjalnie budowane z myślą o tym, aby je wykorzystywać w SPECOLU. Wymaga się jedynie odpowiedniego formatowania przekazywanych informacji — realizuje to tzw. moduł użytkownika [1], opracowywany przez programistę wspólnie z użytkownikiem.

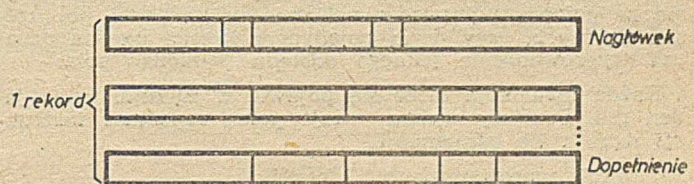
Do informacji zawartej w zbiorze użytkownik odwołuje się tylko przez nazwy pól. Struktura danych, którą określa się dla SPECOLU, może być różna od istniejącej w zbiorze. Rekord w SPECOLU nie musi odpowiadać jednemu zapisowi przetwarzanej kartoteki.

Z punktu widzenia języka zbiór składa się z szeregu zapisów (rekordów) o takiej samej strukturze i w zależności od ich budowy rozróżnia się cztery typy zbiorów.

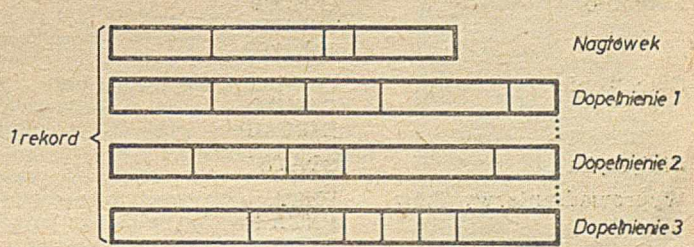
- **Zbiory typu JEDEN** — zbudowane są z rekordów o identycznej strukturze, przy czym rekord zawiera tylko jedno pole grupowe, które nazywane jest nagłówkiem.



- **Zbiory typu DWA** — każdy rekord składa się z nagłówka i jednego lub więcej dodatkowych pól grupowych o takiej samej budowie, które nazywane są dopełnieniem.

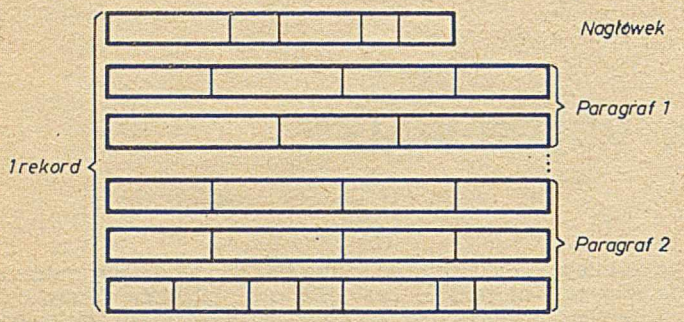


- **Zbiory typu TRZY** — każdy rekord składa się z nagłówka i więcej niż jednego dopełnienia o różnej budowie.



Mgr Andrzej **KUKUŁA** ukończył Wydział Matematyki, Fizyki i Chemii Uniwersytetu Śląskiego (specjalność: matematyka numeryczna) w roku 1972. Pracował w Centrum Informatyki i Badań Ekonomicznych Hutnictwa, odbył szkolenie w firmie ICL w Wielkiej Brytanii. Obecnie pracuje w Zakładzie Problemów Organizacji i Zarządzania Polskiej Akademii Nauk na stanowisku specjalisty-projektanta.

● **Zbiory typu CZTERY** — każdy rekord składa się z nagłówka i szeregu różnego typu dopełnień grupowanych w zestawy, które nazywane są paragrafami.



INSTRUKCJE JĘZYKA SPECOL

Każde żądanie (program) w języku SPECOL zawiera następujące części:

● Instrukcje wprowadzające, w których określa się sposób przetwarzania kartoteki oraz deklaruje się, jakiego fragmentu kartoteki ma to przetwarzanie dotyczyć

● Instrukcje przeszukiwania, zawierające podstawowe deklaracje dotyczące przeszukiwania kartoteki.

Kryterium przeszukiwania może być wystąpienie w polu określonej wartości, obecność pewnej kombinacji znaków, zakres wartości danej itp.

Instrukcja przeszukiwania składa się z symbolu instrukcji, po którym występuje kryterium. Na przykład:
TYPA MIEJSCE (KATOWICE OR BYTOM) AND CENA (<55 000)

Taka instrukcja skierowana do kartoteki nieruchomości pozwala wyszukać wszystkie zapisy, które w polu o nazwie MIEJSCE mają wartość KATOWICE lub BYTOM i jednocześnie ich cena nie przekracza 55 000 zł.

Język SPECOL zawiera kilka odmian instrukcji przeszukiwania, ale zasadniczo ich działanie będzie takie samo, jak w podanym przykładzie. Wiele różnych sposobów podawania wykazu wartości dla określonej danej (pola) pozwala na różnorodne specyfikowanie warunków przeszukiwania.

● Instrukcje zliczania i sumowania

Zliczanie i sumowanie mogą się odbywać na zapisach wybranych do przetwarzania przez wcześniej podane instrukcje przeszukiwania lub, w przypadku ich braku, we wszystkich zapisach w kartotece.

● Instrukcje wyprowadzania wyników

Sposób pisania tych instrukcji jest prosty. Dotyczą one tylko wyprowadzania danych z wybranych rekordów, ponieważ wyniki działań instrukcji zliczania i sumowania są wyprowadzane automatycznie. Pola, których wartości mają być wydrukowane, określa się przez podanie ich nazw. Natomiast rozmieszczenie tabulogramu programuje się za pomocą wprowadzanych między nazwami pól kresek ukośnych, przy czym pojedyncza kreska ukośna oznacza wyprowadzenie jednego odstępu pomiędzy pola.

Każda instrukcja może się pojawić w żądaniu tylko jeden raz. Poniższy wykaz podaje pełen repertuar instrukcji wraz z ich odmianami.

Instrukcje wprowadzające	MODE LMTS
Przeszukiwanie	TYPA TYPB TYPBX TYPBO
Zliczanie	OVCNT INCNT INCNTP
Sumowanie	OVSUM INSUM INSUMP
Statystyka	STATS
Wyprowadzanie wyników	PNTA PNTA+ PNTB
Spacjowanie wierszy	SPCE NOSP
Koniec żądania	END

Każde żądanie musi rozpoczynać się od instrukcji MODE, a kończyć instrukcją END. Wszystkie inne mogą lecz nie muszą wystąpić w żądaniu. Instrukcje STATS, SPCE, NOSP i END nie mają żadnych argumentów.

Oprócz znajomości samego języka SPECOL użytkownik powinien znać również słownik nazw pól, które przyporządkowane zostały każdej danej.

Rozpatrzmy nieco rozbudowany przykład. Wyobraźmy sobie kartotekę nieruchomości, z której chcemy wydrukować dane dotyczące wszystkich domów typu willa, o czterech pokojach, z dużym ogrodem (ponad 400 m²) i garażem, zlokalizowanych w Katowicach lub Bytomiu, o wartości 400—500 tys. złotych.

Przyjmijmy, że struktura jednego zapisu w kartotece jest następująca:

NRZAPISU	MIEJSCE	ULICA	NRDOMU	CENA	ILPOKOI	GARAZ	TYP	UWAGI

Informacje potrzebne do przeszukiwania kartoteki znajdują się w polach MIEJSCE, TYP, CENA,, IL. POKOI, GARAŻ i OGRÓD. Instrukcja, za pomocą której będziemy przeglądać zapisy, będzie miała postać:

TYPA MIEJSCE (KATOWICE OR BYTOM) AND TYP (WILLA) AND CENA (400 000 TO 500 000) AND IL POKOI (4) AND GARAŻ AND OGRÓD (>400)

Łącznik AND, jakim rozpoczęliśmy kolejną część instrukcji, działa jako kontynuacja raz podanego symbolu instrukcji TYPA. Po nazwie pola GARAŻ nie ma wykazu wartości danej, ponieważ to pole zawiera spacje w wypadku, gdy garażu nie ma i taka jest konwencja języka SPECOL. Znak „>" użyty w wyrażeniu oznacza, że wartość pola musi być większa od 400. Ponieważ nie wymagamy tu zliczania ani sumowania, bezpośrednio po tej instrukcji wystąpi wyprowadzenie wyników. Możemy wydrukować wszystkie lub niektóre informacje użyte w instrukcji przeszukiwania, a także inne zawarte w odnalezionym rekordzie, np. adresy nieruchomości.

Instrukcja drukowania może więc mieć postać:

PNTA MIEJSCE/ /ULICA/ /NRDOMU/ /CENA/ /OGRÓD

Nie wyprowadzamy już takich informacji, jak TYP, ILPOKOI i GARAŻ, ponieważ treść ich jest wiadoma (z warunku wyboru). Łącznie całe żądanie będzie miało następującą postać:

```
MODE 1 [1])
TYPA MIEJSCE (KATOWICE OR BYTOM) AND
TYP (WILLA) AND CENA (400 000 TO 500 000) AND
ILPOKOI (4) AND GARAŻ AND OGRÓD (>400)
PNTA MIEJSCE/ /ULICA) /NRDOMU/ /CENA/ /O-
GRÓD END
```

Wyniki tak zdefiniowanego żądania będą na przykład następujące:

KATOWICE	POLNA 41	420000	1200
KATOWICE	JACKA 5	482000	800
BYTOM	KOŚCIUSZKI 49	410000	750

Z wydruku widzimy, że tylko trzy zapisy w kartotece spełniły warunki przeszukiwania.

WDRAŻANIE JĘZYKA SPECOL

Pakiet języka SPECOL jest stosowany na komputerach ICL System 4, których w Polsce jest kilka. Pakiet składa się z dwu modułów, z których jeden to kompilator żądań, a drugi to moduł, który należy dopasować do konkretnej kartoteki oraz sposobu wprowadzania żądań zapisanych w SPECOLU i sposobu wyprowadzania wyników. Drugi moduł, zwany modułem użytkownika, zawiera podprogramy wejścia/wyjścia i słownik nazw danych. Modularne rozwiązanie zapewnia prostotę, z jaką można modyfikować sposób wprowadzania żądania i wyprowadzania wyników.

[1]) Zwrot MODE określa typ zbioru; tu zakładamy, że przeszukujemy jedynie nagłówki.

Modyfikacją samego modułu użytkownika zajmuje się personel specjalistyczny (programista), który udostępnia użytkownikowi słownik nazw pól oraz strukturę danych przedstawioną najczęściej w sposób graficzny. Przy każdej z nazw programista powinien podać również ważne dla użytkownika informacje o polu, na przykład czy można sumować jego wartość lub ile znaków może maksymalnie zawierać pole znakowe.

Ponieważ modyfikowanie pakietu SPECOL jest dla doświadczonego programisty dość proste, a czas tworzenia żądania nie powinien przekraczać paru minut, język może być stosowany jako narzędzie do bezpośredniej pracy na komputerze w trybie konwersacyjnym.

Wadą języka SPECOL jest czas oczekiwania na odpowiedź (przeszukiwanie sekwencyjne), niemniej z uwagi na uniwersalność i prostotę rozwiązania może stanowić nieocenione narzędzie przy wdrażaniu systemu teletransmisji. Czas odpowiedzi na żądanie specolowe jest uzależniony oczywiście od wielkości zbioru oraz od zastosowanego nośnika pamięci. W przypadku gdy kartoteka znajduje się na taśmie magnetycznej czas odpowiedzi będzie limitowany czasem przewijania taśmy. Lepsze wyniki osiąga się przy stosowaniu dysków magnetycznych.

OCENA JĘZYKA SPECOL

To, że dzięki swej nieskomplikowanej strukturze SPECOL może być stosowany przez użytkownika (nieinformatyka), stanowi chyba jego największą zaletę.

Pisaniem żądań w języku SPECOL rządzi kilka reguł, które nawet przy niewielkim doświadczeniu są łatwe do zrozumienia. Zakres żądań, które mogą być stawiane, obejmuje żądania proste, takie jak zwięzłe wyciągi informacji dla szczebla kierowniczego oraz pytania bardziej złożone, jak na przykład dotyczące identyfikacji obiektu na podstawie niepełnej informacji czy wykrywania nowych powiązań między danymi.

Bardzo wygodne jest korzystanie z języka SPECOL w trybie bezpośredniej komunikacji użytkownika z komputerem.

O zaletach języka świadczyć może bieżące korzystanie z tego pakietu w Z. H. Cegielski w Poznaniu, przemyśle okrętowym i ostatnio w branży hutniczej. Ponadto konkretne zastosowanie pakiet SPECOL znalazł w Przedsiębiorstwie Handlu Zagranicznego „Stalexport" w Katowicach. Służy on tam do przeszukiwania kartotek eksploatowanych w tym przedsiębiorstwie. Są to kartoteki dostaw importowych, zamówień oraz kontraktów. SPECOL jest eksploatowany w dwu wersjach: standardowej, gdzie żądanie wprowadza się z kart perforowanych, a wyniki drukowane są na drukarce, oraz w wersji konwersacyjnej. W Centrum Informatyki i Badań Ekonomicznych Hutnictwa zastosowano SPECOL również w wersji konwersacyjnej poprzez monitor klawiaturowo-ekranowy ICL 7181.

LITERATURA:

[1] M. Kopański, A. Kukuła, J. Muzyczyszyn: SPECOL — podręcznik programowania (wydanie II rozszerzone i zmienione). Centrum Informatyki i Badań Ekonomicznych Hutnictwa, Katowice 1977

[2] Technical Publication Service ICL, SPECOL. Wydanie I, 1972

BOLESŁAW ŚLIZ

Ośrodek Organizacji i Informatyki Przemysłu Petrochemicznego
PETROINFORM
Kraków

Problemy projektowania zbioru branżowego

W miarę rozwoju systemów informatycznych na szczeblu zjednoczeń i resortów na coraz większą skalę wykorzystuje się w tych systemach tzw. zbiory branżowe i resortowe, tj. zbiory zawierające informacje na temat działalności co najmniej kilku jednostek gospodarczych branży czy resortu. W zasadzie informacje w zbiorze branżowym powinny dotyczyć wszystkich jednostek gospodarczych branży, lecz w praktyce nie zawsze jest to możliwe i celowe. W artykule przedstawiono doświadczenia zebrane w toku projektowania, wdrażania i eksploatacji zbioru dostawców i odbiorców Zjednoczenia Przemysłu Rafineryjnego i Petrochemicznego.

PRZEBIEG PROCESU PROJEKTOWANIA

Potrzeba utworzenia branżowego zbioru dostawców i odbiorców była ściśle związana ze specyfiką projektowania i wdrażania systemów informatycznych w Ośrodku Organizacji i Informatyki Przemysłu Petrochemicznego PETROINFORM[1]). W Ośrodku obowiązuje bowiem zasada, że każdy zaprojektowany system powinien być powielony w każdym przedsiębiorstwie Zjednoczenia. Zasada ta narzuca konieczność stosowania m.in.:

a) zunifikowanych form dokumentów wejściowych

b) różnorodnych form maszynowych nośników danych (ze względu na różne rodzaje urządzeń do przygotowania danych)

c) jednolitych zasad symbolizacji

d) wspólnych zbiorów indeksowych dla wszystkich systemów eksploatowanych w Ośrodku.

[1]) Bliższe omówienie działalności PETROINFORMU w INFORMATYCE nr 12 z 1975 r.

Zbiór dostawców i odbiorców przedsiębiorstw zgrupowanych w Zjednoczeniu PETROCHEMIA (zwany dalej „Branżowym Indeksem Kontrahentów", w skrócie BIK) jest zbiorem indeksowym, a więc zawierającym te informacje o kontrahentach, które mają względnie stały charakter. Podstawowa trudność prawidłowego zaprojektowania zawartości zbioru polega na tym, że wykorzystywany on ma być nie tylko w systemach równolegle z nim projektowanych, ale również w systemach przewidzianych do opracowania w okresie późniejszym. Niezwykle trudne jest bowiem sprecyzowanie potrzeb informacyjnych systemu, dla którego nie ma jeszcze w pełni określonego zakresu tematycznego. W związku z tym autorzy projektu zmuszeni

Mgr Bolesław ŚLIZ jest absolwentem Wyższej Szkoły Ekonomicznej w Krakowie. Od 1969 r. pracował w ZETO Kraków, później (od 1971 r.) w Ośrodku PETROINFORM. Zajmuje się głównie systemami informatycznymi w obrocie towarowym. Jest autorem lub współautorem kilku eksploatowanych systemów informatycznych z tej dziedziny. Jest także wykładowcą na Podyplomowym Studium Projektowania SEPD przy Akademii Ekonomicznej w Krakowie z przedmiotu „Projektowanie systemów informatycznych w przedsiębiorstwach handlowych".

byli zastosować sposób projektowania odmienny od najczęściej używanej w PETROINFORMIE metody BISAD [2]), rozpoczynającej projektowanie od określenia celów systemu. Sposób ten polega na ustaleniu ostatecznej struktury zbioru w trzech fazach.

W fazie pierwszej określone zostały możliwie wszystkie cechy, jakie mogą charakteryzować badany obiekt (kontrahenta). Jedynym ograniczeniem była względna stałość cechy. Stąd też odrzucone zostały takie cechy, jak: wielkość i rodzaj produkcji, powiązania kooperacyjne, dane o zatrudnieniu itp. W rezultacie otrzymano dość obszerny zestaw cech, z których należało wytypować te, które mogą być wykorzystane przez branżowe systemy informatyczne. Wyboru tego dokonano w dwóch następnych fazach.

W fazie drugiej zebrane zostały potrzeby informacyjne ze strony systemów informatycznych projektowanych w PETROINFORMIE. Potrzeby informacyjne określone zostały przez głównych projektantów wszystkich kompleksów tematycznych realizowanych w Ośrodku. Suma logiczna tych potrzeb utworzyła zespół cech niezbędnych.

W fazie trzeciej przeprowadzono analizę wszystkich pozostałych cech i wyodrębniono spośród nich takie, które mogą okazać się przydatne dla systemów informatycznych, przewidzianych do zrealizowania w przyszłości.

Opisana wyżej metoda postępowania doprowadziła do wytypowania następujących cech, niezbędnych do wprowadzenia do zbioru „Branżowy Indeks Kontrahentów":

— identyfikator wg systemu REGON (7-cyfrowy)
— symbol klasyfikacyjny wg systemu REGON (19-cyfrowy)
— identyfikator dodatkowy (2-cyfrowy)
— nazwa kontrahenta
— adres (wieś lub ulica, numer domu)
— kod pocztowy
— miejscowość (nazwa)
— skrót nazwy kontrahenta
— symbol płatnika
— nazwa banku
— kod pocztowy banku
— adres banku
— numer konta bankowego działalności podstawowej
— nazwa stacji kolejowej
— numer stacji.

Jako naczelną zasadę przyjęto, że symbolizacja branżowa kontrahentów musi być zgodna z systemem identyfikacji i klasyfikacji jednostek gospodarki narodowej — REGON. Wprowadzenie tzw. identyfikatora dodatkowego spowodowane zostało faktem adresowania niektórych dostaw do odbiorców nie będących odrębną jednostką gospodarki narodowej w nomenklaturze systemu REGON (chodzi tu głównie o bazy, filie, zespoły magazynów itp.) [3]).

Pewnym problemem było również wprowadzenie do zbioru informacji o płatniku. Występują tu trzy przypadki. Pierwszy, najczęściej spotykany, polega na tym, że odbiorca dostawy jest równocześnie płatnikiem. W drugim przypadku odbiorca otrzymuje dostawy, natomiast wszystkie rachunki płaci inna jednostka, zwykle jednostka nadrzędna. Ma to miejsce zwłaszcza w odniesieniu do jednostek działających na zasadach tzw. niepełnego wewnętrznego rozrachunku. Trzeci przypadek jest mieszany, niektóre dostawy dla danego odbiorcy są finansowane przez niego, inne zaś przez inną jednostkę gospodarczą. Taka sytuacja występuje bardzo rzadko, np. w obrocie niektórymi produktami rafineryjnymi. Autorzy projektu uznali za niecelowe wprowadzanie informacji o kilku płatnikach dla jednego kontrahenta, gdyż wymagałoby to znacznego rozbudowania wszystkich rekordów w zbiorze, chociaż informacja ta jest istotna tylko w odniesieniu do kilkunastu kontrahentów. Przyjęto tu rozwiązanie polegające na użyciu kilku symboli kontrahenta (w oparciu o tzw. identyfikator dodatkowy) jako bardziej ekonomiczne.

---

[2]) BISAD — Business Information Systems. Analysis and Design, prognostyczna metoda projektowania, opracowana przez firmę HONEYWELL. Metoda ta jest stosowana w PETROINFORMIE w wersji zmodyfikowanej przez zespół pracowników Ośrodka.

[3]) W ramach opracowanego w PETROINFORMIE systemu „Realizacja Produkcji i Zbytu" odbywa się m.in. fakturowanie dostaw. Wymagany przepisami (OWS) stopień szczegółowości informacji drukowanych na fakturze jest większy niż to przewiduje system REGON. Podobna sytuacja ma miejsce w systemie dystrybucji nawozów przy drukowaniu zamówień.

Po wyodrębnieniu wszystkich cech okazało się, że siedem spośród nich (1, 2, 4, 5, 6, 7, 8) są identyczne z tymi, które stosuje Główny Urząd Statystyczny w założonym przez siebie zbiorze jednostek gospodarki narodowej wg systemu REGON (zbiór o nazwie mnemonicznej RG-UA-OGOLNY). Ponieważ Ośrodek Badawczo-Rozwojowy Systemu Państwowej Informacji Statystycznej GUS oferuje możliwość kopiowania taśm magnetycznych zbiorów systemu REGON — zdecydowano wykorzystać tę drogę do założenia zbioru „Branżowy Indeks Kontrahentów", pomimo braku kompatybilności maszyn ODRA oraz HONEYWELL.

TECHNOLOGIA PRZETWARZANIA

Po zdefiniowaniu zawartości zbioru opracowanie technologii przetwarzania nie stwarzało większych trudności. Specyfika „Branżowego Indeksu Kontrahentów" wymaga, żeby:
— zbiór był o dostępie bezpośrednim (nie sekwencyjnym)
— zapewniał łatwość aktualizacji
— uniemożliwiał aktualizację w sposób niewłaściwy lub przez osoby nieuprawnione.

Zapewnienie realizacji pierwszych dwóch warunków nie wymaga bliższych wyjaśnień. Zbiór został założony na dysku magnetycznym, organizacja zbioru jest indeksowo-sekwencyjna. Aktualizacja zbioru jest możliwa zarówno poprzez wprowadzanie nowych rekordów w całości, jak i poprzez zmianę wartości dowolnej cechy. Spełnienie warunku trzeciego było bardziej kłopotliwe, bowiem z „Branżowego Indeksu Kontrahentów" w PETROINFORMIE korzystają systemy informatyczne eksploatowane przez przedsiębiorstwa branży petrochemicznej w całej Polsce. Użytkownicy tych systemów muszą mieć prawo nie tylko do korzystania ze zbioru w sensie pobierania z niego danych, lecz również do aktualizacji danych w zbiorze. W dodatku aktualizacja musi być przeprowadzona możliwie szybko, zwłaszcza w przypadku zmiany konta bankowego, czy też zmiany stacji odbierającej przesyłki. Z kolei aktualizacja przeprowadzona przez jednego z użytkowników może być niewłaściwa z punktu widzenia innych użytkowników, np. zmiana stacji kolejowej może dotyczyć tylko wybranego asortymentu, a nie wszystkich towarów wysyłanych do danego odbiorcy.

Skutecznym rozwiązaniem tego problemu byłoby utworzenie w Ośrodku specjalnego zespołu do spraw eksploatacji BIK, którego zadaniem powinna być m.in. weryfikacja danych do aktualizacji zbioru, przysyłanych przez użytkowników. Każda próba wprowadzenia do zbioru danych formalnie poprawnych, lecz nieprawidłowych merytorycznie mogłaby być wyjaśniona przez pracowników zespołu. Drobne błędy poprawiano by od razu, inne poważniejsze rozbieżności — po ich wyjaśnieniu z użytkownikiem. W ten sposób zahamowano by tendencję do wprowadzania do zbioru następnych rekordów, odnoszących się do tego samego kontrahenta, które w ogóle nie różnią się między sobą lub różnica jest nieznaczna.

Ponieważ opisanego rozwiązania nie można było zrealizować ze względów kadrowych, zastosowano inną drogę postępowania. Każdy rekord zawiera symbol jednostki autorskiej, którą jest użytkownik zapisujący dane o kontrahencie jako pierwszy. Jednostka autorska jest odpowiedzialna za prawidłowość danych w rekordzie. Utworzono zabezpieczenie programowe uniemożliwiające aktualizację danych w rekordzie przez innego użytkownika. Każda próba takiej aktualizacji jest traktowana jako błędna, a na raporcie błędów użytkownik otrzymuje informację kto jest jednostką autorską i ma prawo aktualizować dany rekord. PETROINFORM ma połączenia teletransmisyjne z większością użytkowników systemów informatycznych eksploatowanych w Ośrodku, zatem wszystkie te działania nie powodują poważniejszych opóźnień.

Kilka słów wyjaśnienia wymaga sposób posługiwania się zbiorem RG-UA-OGOLNY, kopiowanym w OBR SPIS Głównego Urzędu Statystycznego. Ponieważ komputery ODRA, którymi dysponuje GUS, nie są kompatybilne z komputerem HONEYWELL 3200 PETROINFORMU, zatem konieczne stało się sporządzenie specjalnego programu tłumaczącego zapis na taśmie magnetycznej otrzymanej z GUS na zapis w kodzie USASCII. Program został napisany w języku asemblerowym (EASYCODER), a jego zasadniczą funkcją jest zmiana kodów. Ważne jest, że nie napotkano tu na problemy natury sprzętowej, bowiem parametry techniczne taśmy, a w szczególności gęstość zapisu, wielkość przerw międzyblokowych i liczba ścieżek, odpowiadały wymogom komputera H-3200.

Ze zbioru RG-UA-OGOLNY po przetłumaczeniu wybierane są dane tylko z tych rekordów, dla których przygotowane są jako odrębny zbiór dane uzupełniające (numery kont bankowych, stacja kolejowa itp.). Chodzi o to, że zbiór BIK liczyć będzie docelowo nie więcej niż 20 tys. rekordów, zaś zbiór RG-UA-OGOLNY ma mieć ok. 100 tys. rekordów i wprowadzenie wszystkich tych rekordów do zbioru branżowego nie byłoby celowe.

Barierą ograniczającą możliwość korzystania ze zbioru RG-UA-OGOLNY okazał się czas przetwarzania. Pełny przebieg maszynowy, składający się z programu tłumaczącego, programu dopisującego dane uzupełniające i programów alokacji i ładowania danych na dysk magnetyczny, trwa kilka godzin. W związku z tym opisana droga postępowania wykorzystywana jest bardzo rzadko. Dopiero przy dopisywaniu do zbioru BIK rekordów w ilości rzędu kilku tysięcy uzyskuje się oszczędności czasu przygotowania danych (polegające na pomijaniu perforacji nazw, adresów, symboli klasyfikacyjnych), równoważące wydłużenie czasu przetwarzania. Zwykle do zbioru dopisuje się jednorazowo kilkadziesiąt lub rzadziej kilkaset rekordów; wówczas na maszynowym nośniku danych nanosi się komplet informacji o kontrahencie i aktualizacja odbywa się bez pośrednictwa zbioru RG-UA-OGOLNY. Czas przetwarzania wynosi w tym przypadku kilkanaście minut. Obowiązkiem użytkownika wprowadzającego dane jest zachowanie pełnej zgodności symbolizacji i nazewnictwa z symbolizacją i nazewnictwem stosowanymi w systemie REGON i publikowanymi w „Wykazie jednostek gospodarki uspołecznionej oraz ich numerów statystycznych".

EKSPLOATACJA ZBIORU

Informacje o kontrahentach zawarte w zbiorze BIK są dostępne dla każdego użytkownika systemów informatycznych przetwarzanych w Ośrodku PETROINFORM. Większość systemów posługuje się zbiorem w celu wydrukowania na tabulogramach nazwy kontrahenta lub posortowania danych według cech określonych w symbolu klasyfikacyjnym. W największym zakresie i najczęściej korzystają ze zbioru system „Realizacja Produkcji i Zbytu", w ramach którego odbywa się fakturowanie dostaw, oraz system „Ewidencja i rozliczenie dostaw nawozów mineralnych", w którym odbywa się drukowanie zleceń (zamówień).

Ponadto raz na kwartał sporządza się i rozsyła do wszystkich użytkowników tabulogram aktualnej zawartości zbioru. Często wykonuje się również tabulogramy o układzie żądanym przez użytkownika. Program wydawniczy ma bowiem możliwość drukowania pełnej zawartości zbioru, uporządkowanie według dowolnej cechy, jak również drukowania tylko niektórych rekordów spełniających zadane warunki. Interesujące jest tu częste żądanie wydruku informacji o kontrahentach podległych wskazanym resortom czy zjednoczeniom, w kolejności alfabetycznej według nazw miejscowości — siedzib kontrahenta. Układ taki zapewniać ma łatwość posługiwania się wydrukiem w działach zbytu i zaopatrzenia przedsiębiorstw.

Aktualizacja zbioru odbywa się co 2—3 dni, a w razie potrzeby częściej. Aktualizować dane może każdy użytkownik, ale zawsze za pośrednictwem jednostki autorskiej.

Obecnie zbiór zawiera dane o 6 tysiącach kontrahentów, w roku bieżącym liczba ta powinna ulec podwojeniu.

ZBIGNIEW ODROWĄŻ-SYPNIEWSKI
Instytut Informacji Naukowej, Technicznej i Ekonomicznej
Warszawa

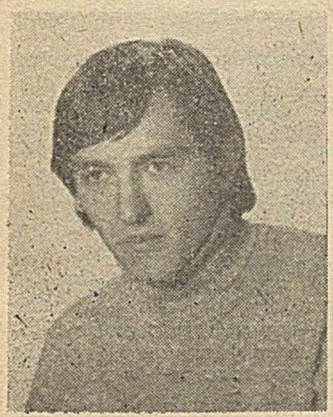
MICHAŁ OSTROWSKI
Ośrodek Informatyki Technicznej i Przetwarzania Danych w OLPiT
Warszawa

SYWIR—system wyszukiwania informacji i redagowania biuletynów

W związku z prowadzonymi w kraju pracami nad zautomatyzowaniem ośrodków sieci INTE oraz ze względu na przewidywane wykorzystanie do tego celu maszyn Jednolitego Systemu w Instytucie Informacji Naukowej, Technicznej i Ekonomicznej w Warszawie opracowano uogólniony system wyszukiwawczo-wydawniczy, dostosowany do tych maszyn. Założono, że system uwzględniać powinien:

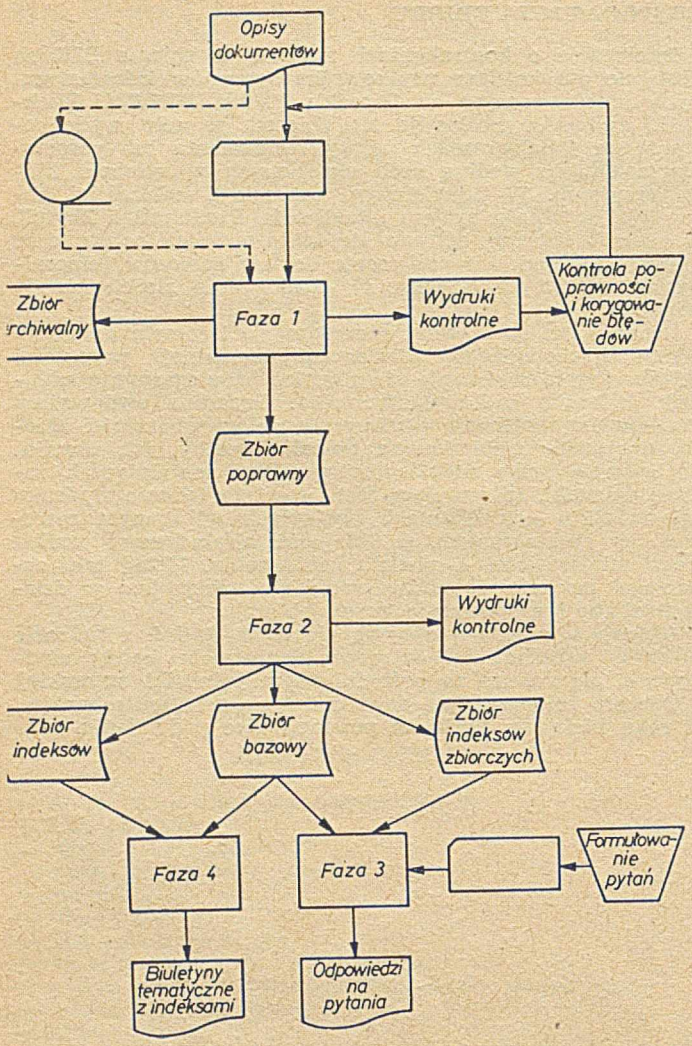
— znaczny stopień dowolności postaci danych
— możliwość wyszukiwania informacji według dowolnie wybranego kryterium
— redagowanie dowolnych biuletynów tematycznych wraz z odpowiednimi indeksami
— korzystanie z języka wyszukiwawczego o prostej strukturze
— dostosowanie do działania na maszynach Jednolitego Systemu lub IBM 360/370 pod kontrolą systemów operacyjnych DOS i OS.

Ogólny schemat systemu zrealizowanego w Instytucie INTE w ramach problemu międzyresortowego SINTO przedstawiono na rysunku.



Mgr inż. Zbigniew ODROWĄŻ-SYPNIEWSKI ukończył studia w 1972 r. na Wydziale Elektroniki PW. Zajmuje się systemami informacyjnymi oraz badaniami poprawności programów.

Mgr inż. Michał OSTROWSKI ukończył studia w 1972 r. na Wydziale Elektroniki PW. Zajmuje się organizacją baz danych i wyszukiwania na potrzeby informacji telefonicznej.



Ogólny schemat systemu

DANE WEJŚCIOWE

Danymi wejściowymi systemu SYWIR mogą być przykładowo: opisy bibliograficzne druków zwartych, ciągłych, norm, patentów, notek prasowych, haseł encyklopedycznych, prac naukowo-badawczych, sprawozdań z podróży, materiałów konferencyjnych, tłumaczeń itp.

Ze względu na zróżnicowaną treść wspomnianych opisów przyjęto ich podział na tzw. paragrafy. Paragrafy te w zależności od potrzeby mogą być przypisywane różnym tematycznie informacjom, takim jak: autor, tytuł, słowa kluczowe, sygnatura, klasyfikacja, promotor, abstrakt, data, koszt, itp. Paragrafy oznaczane są trzycyfrowymi numerami umieszczonymi w rekordzie danych wejściowych jako 9, 10 i 11 znak. Oprócz wspomnianego pola numeru paragrafu format wejściowy danych zawiera jeszcze pole, w którym umieszczony jest numer rejestracyjny opisu dokumentu (znaki 1—8) oraz pole treści (znaki 12—80).

Stosowanie paragrafów zapewnia praktycznie nieograniczoną elastyczność w zakresie podziału i interpretacji poszczególnych elementów treści danych wejściowych. W realizowanych w Polsce systemach przetwarzania danych tekstowych występują istotne trudności, wynikające z możliwości wyprowadzania liter polskich (ą, ć, ę, ł, ń, ó, ś, ź, ż). Brak tych liter w dokumentach wynikowych (np. biuletynach) powoduje niedopuszczalne zniekształcenie tekstów szczególnie rażące np. w przypadku nazwisk.

W systemie SYWIR przyjęto więc dla celów przygotowania danych dwuznakowe kody symbolizacji liter polskich według następującej reguły:

Ą — A2
Ć — C1
Ę — E2
Ł — L4
Ń — N1
Ó — O1
Ś — S1
Ź — Z1
Ż — Z3

Zasada powyższa zachowuje rosnący porządek alfabetyczny, co w przypadku sortowania wyrazów ustawia np.: literę Ą po literze A, gdy występują one na tych samych pozycjach. Jest to osiągalne również w rozwiązaniach wprowadzających bezpośrednio litery polskie poprzez wymianę na litery polskie niektórych rzadko używanych znaków specjalnych w łańcuchu drukarki wierszowej lub matryc w drukarkach mozaikowych.

Przyjęta zasada kodowania umożliwia wypisywanie liter polskich przez drukarki wierszowe nie posiadające odpowiednich czcionek metodą podwójnego (nakładającego się) druku (znak podstawowy + znak diakrytyczny). Podobnie jak wprowadzenie w systemie SYWIR liter polskich przewidziano również możliwość rozróżnienia liter małych i dużych. Rozróżnienie to wprowadzono ze względu na przewidywane wykorzystanie w systemie fotoskładu jako urządzenia wyjściowego. W rozwiązaniu tym pojedyncza litera interpretowana jest jako litera duża, gdy przygotowując dane wejściowe poprzedzamy ją mało wykorzystywanym u nas znakiem $. Jeżeli cały ciąg kolejnych liter jednego wyrazu ma być interpretowany jako ciąg liter dużych, to należy poprzedzić go dwoma znakami $.

Dane wejściowe mogą być przygotowywane na kartach dziurkowanych lub bezpośrednio na taśmach magnetycznych za pomocą rejestratorów klawiaturowych.

STRUKTURA I DZIAŁANIE SYSTEMU

W działaniu systemu SYWIR wyodrębnić można następujące cztery fazy:

1 — zakładanie i aktualizowanie bazy danych
2 — tworzenie zbiorów pomocniczych
3 — wyszukiwanie informacji
4 — redagowanie biuletynów.

Oprogramowanie

Programy systemu zostały napisane w trzech językach: ASSEMBLER, FORTRAN oraz PL/1, w wersjach korzystających z systemu operacyjnego DOS/JS albo OS/IBM. Przyjęto koncepcję programowania modularnego, która odzwierciedla naturalny podział funkcji systemu na wspomniane cztery fazy. Programy poszczególnych faz komunikują się między sobą poprzez zbiory o ustalonych strukturach rekordów. Umożliwia to dołączanie programów dodatkowych realizujących specyficzne wymagania użytkownika, a także równoległą pracę osób piszących programy.

Faza 1

Faza 1 realizuje zakładanie, korygowanie i aktualizację bazy danych. W trakcie tej fazy dokonywana jest formalna i merytoryczna kontrola poprawności wprowadzanych danych.

Kryteria kontroli mogą być zmienione w zależności od zastosowania systemu. Najczęściej stosowanymi kryteriami są:

— kompletność wprowadzanych danych (opisów dokumentów), np. występowanie paragrafów obligatoryjnych, początku i końca opisu itp.
— nie powtarzanie się tych samych paragrafów w ramach jednego dokumentu i dokumentu w ramach bazy
— poprawność wypełnienia pól sformatowanych (np.: dopuszczalny kod pola klasyfikacji, daty)
— poprawność wzajemnego występowania lub niewystępowania pewnych skojarzonych paragrafów w dokumencie (np. paragraf zawierający dane o promotorze musi zawsze występować w opisie pracy doktorskiej)
— kontrola słownictwa wskazanych paragrafów (np. występowania słów kluczowych zawartych w odpowiednich słownikach).

Karty korygujące wykryte błędy mogą:
— usunąć cały dokument
— usunąć lub usunąć i zastąpić treść wskazanego paragrafu dokumentu
— wstawić treść brakujących paragrafów pomiędzy wskazane paragrafy dokumentu.

Czynność korygowania błędów połączona jest z jednoczesnym ponownym badaniem poprawności dokumentów. W wyniku każdorazowego badania wyprowadzany jest obszerny komunikat informacyjno-statystyczny, którego postać uzależniona jest od przyjętych kryteriów kontroli.

Faza 2

W fazie 2 odbywa się posortowanie dokumentów na działy tematyczne na podstawie wskazanej klasyfikacji (UKD, rubrykatora MCINT, sygnatury itp.) oraz założenie zbiorów indeksów tematycznych, służących jednocześnie jako zbiory odwrócone w procesie wyszukiwania. W trakcie procesu przetwarzania dokumenty uzyskują dogodny do dalszego przetwarzania format, a mianowicie:

— zostają rozszerzone o numer systemowy (rok wprowadzenia do bazy, numer serii tematycznej, numer zeszytu biuletynu danej serii i numer opisu w ramach danego biuletynu okresowego)

— zostają usunięte z nich informacje zbędne w dalszym procesie wydawniczym i wyszukiwawczym.

Dla opisów dokumentów przydzielonych do kilku różnych działów tematycznych tworzony jest osobny zbiór wzajemnych odsyłaczy. Otrzymany sekwencyjny zbiór dokumentów o nowym formacie jest wykorzystywany do zredagowania treści biuletynów, utworzenia zbiorów indeksów oraz jako baza informacyjna w procesie wyszukiwania.

Zbiory indeksów (autorskiego, przedmiotowego, instytucji itp.) tworzone są na podstawie wskazanych paragrafów. Jednocześnie z tworzeniem zbiorów indeksów do danego numeru biuletynu uzupełniane są odpowiednie indeksy zbiorcze, dotyczące dłuższego okresu czasu. Indeksy te dzięki zawieraniu numeru systemowego, umożliwiającego jednoznaczne zaadresowanie dokumentu, są podstawą do prowadzenia procesu wyszukiwania.

Faza 3

Faza 3 obejmuje wyszukiwanie informacji, która jest realizowana w trybie wsadowym. Pytania wprowadzane na kartach dziurkowanych analizowane są pod względem poprawności syntaktycznej oraz rozbicia na podgrupy (fasety). Fasetą jest zawarty w pytaniu zbiór słów kluczowych wzajemnie ze sobą sumowanych logicznie. Fasety wzajemnie są iloczynowane logicznie. Możliwe jest również użycie w pytaniu nawisów oraz operatora negacji logicznej.

Realizacja sumy logicznej odbywa się na zasadzie sumy zbiorów adresów dokumentów, w których napotykane były wskazane słowa kluczowe.

Iloczyn logiczny tworzony jest jako przecięcie zbiorów adresów wspomnianych dokumentów. Negacja logiczna realizowana jest jako różnica zbiorów adresów.

Odpowiedź zawiera wydruk treści pytania, statystykę kolejnych kroków procesu wyszukiwania (liczby dokumentów relewantnych dla poszczególnych słów kluczowych oraz dla operacji logicznej) oraz treść wyszukanych dokumentów.

Uproszczona syntaktyka formułowania pytań

Pytanie należy formułować w języku wzorowanym na stosowanym w systemie ISIS (Integrated Scientific Information System) stworzonym w ILO (International Labour Organization) w Genewie. Struktura pytań w tym języku sprowadza się do następujących zasad:
— pytanie należy poprzedzić numerem identyfikującym osobę zadającą pytanie (system weryfikuje uprawnienia do korzystania z procesu wyszukiwania)
— pytanie zbudowane jest ze słów kluczowych połączonych odpowiednio operatorami
— w pytaniu dopuszcza się stosowanie następujących operatorów:
.= (początek pytania)
.* (operator logiczny „lub")
.+ (operator logiczny „i")
.— (operator logiczny negacji)
.((identyfikator początku fasety)
.) (identyfikator końca fasety)
.$ (operator końca pytania)
.$$ (operator końca procesu wyszukiwania)
— po każdym z operatorów (z wyjątkiem), $ i $$) wolno umieścić tylko jedno słowo kluczowe
— na każdej karcie, zawierającej słowo kluczowe, należy podać numeryczny identyfikator typu indeksu, z którego ono pochodzi (autorskiego, przedmiotowego, instytucji itp.).

Faza 4

Działanie fazy 4 polega na redagowaniu treści biuletynów. Celem jest sformatowanie treści dokumentów i indeksów w postaci gotowej do powielenia. Formatowanie obejmuje podział tekstu wynikowego na kolumny i strony, adiustację wierszy, dodanie nagłówków działów, numerację stron, dołączenie spisu treści itp.

W obecnej wersji system wypisuje biuletyny na drukarce wierszowej, istnieje jednak możliwość jego rozbudowania o moduły tłumaczenia treści na inne kody i dodanie rozkazów sterujących fotoskładem.

WYMAGANIA SPRZĘTOWE SYSTEMU

System SYWIR wymaga następującej minimalnej konfiguracji komputera Jednolitego Systemu:

— pamięć operacyjna 44 K bajtów
— 3 jednostki pamięci dyskowej
— czytnik kart
— drukarka wierszowa.

CECHY ODRÓŻNIAJĄCE SYSTEM SYWIR OD PODOBNYCH SYSTEMÓW

W porównaniu z dotychczas opracowanymi w kraju systemami informacyjno-wyszukiwawczymi system SYWIR wyróżnia się następującymi innowacjami:

— uwzględnieniem liter polskich w wydruku tekstów na drukarkach bez takich liter

— sposobem kodowania wyżej wymienionych liter gwarantującym ich sortowanie we właściwej kolejności

— możliwością rozróżniania liter dużych i małych w przypadku współpracy z fotoskładem

— bogatym repertuarem indeksów
— uwzględnieniem całości problemów wyszukiwawczych i wydawniczych

— możliwością pracy pod kontrolą systemów operacyjnych OS/IBM i DOS/JS

— dużą łatwością obsługi operatorskiej oraz dołączania nowych lub zmiany istniejących modułów oprogramowania.

UWAGI PO WSTĘPNYM OKRESIE EKSPLOATACJI

Eksperymentalna baza danych systemu (1500 opisów dokumentów) stworzona została na podstawie kart informacyjnych o pracach naukowo-badawczych. Oprogramowanie uruchomione na komputerze Jednolitego Systemu R-20 pod kontrolą systemu operacyjnego DOS oraz na komputerze IBM 360/50 pod kontrolą systemu OS.

Realizacją systemu SYWIR, złożonego z 23 programów, zajmowało się łącznie 6 osób przez pół roku. Prostota konstrukcji systemu i koncepcja programowania modularnego umożliwia bardzo szybkie opracowanie i uruchomienie programów oraz ich przenoszenie pomiędzy wyżej wspomnianymi różnymi instalacjami sprzętu komputerowego.

Wielkość bazy danych przeznaczonej do wyszukiwania oczywiście uzależniona jest od pojemności przyłączonych pamięci dyskowych. Korzystne wydaje się przeorganizowanie zbiorów odwróconych i zbioru podstawowego (zastosowanie organizacji bezpośredniego dostępu do zbiorów odwróconych oraz kompresji rekordów zbioru podstawowego). W obecnej chwili na dwóch pakietach dyskowych przechowywana jest baza danych systemu z jej odpowiednimi zbiorami odwróconymi, natomiast biblioteka programów i zbiory robocze zapisane są na trzecim pakiecie.

Przygotowanie danych dla systemu wymaga szczególnej uwagi ze względu na dwuznakowe kody liter polskich, a także rozróżnianie liter dużych i małych (jeśli jest stosowane).

Ze względu na sposób uzyskiwania druku liter polskich (przez podwójny wydruk) konieczne jest szczególnie dokładne wyregulowanie mechanizmów drukarek wierszowych.

LITERATURA:

[1] Meadow Ch. T.: Analiza systemów informatycznych, WNT, Warszawa 1972

[2] Schieber W. D.: Technical manual on ISIS. Statskontoret, Stockholm 1972

[3] Hume J., Holt R.: Structured programing using PL/1 and SP/K. A Prentice-Hall Company, Reston, Wirginia 1975

Jednolity system informatyczny

Spółdzielczość mieszkaniowa, która w 1976 roku obchodziła 20-lecie powojennej działalności, zaś wkrótce potem zameldowała o oddaniu do użytku milionowego mieszkania, ciągle znajduje się w centrum zainteresowania opinii publicznej. Powody tego są znane i oczywiste. Potrzeby mieszkaniowe stale rosną, zaległości i ich przyczyny nie są tajemnicą — źródłem kłopotów są wiadome niedomagania resortu budownictwa. W tej sytuacji kolejka niecierpliwych jest szczególnie zainteresowana modernizacją procesu technologii projektowania i realizacji inwestycji budowlanych, respektowaniem terminowości i sprawiedliwości w rozdziale kluczy do mieszkań.

Nie twierdząc bynajmniej, że informatyka była najważniejszym stymulatorem przyrostu powierzchni mieszkaniowej lub uniwersalnym panaceum na jej sprawiedliwą dystrybucję, sądzimy, że warto zaprezentować udział komputerów w projektowaniu, realizacji i eksploatacji budownictwa mieszkaniowego. Tym bardziej, że w końcu kwietnia br. Centralny Związek Spółdzielni Budownictwa Mieszkaniowego zatwierdził program rozwoju zastosowań informatyki na okres 1978—1985.

Podstawą tego programu jest bilans potrzeb, aktualnych możliwości subwencjonowania informatyki i krytyczna ocena jej dotychczasowego rozwoju w jednostkach wchodzących w skład CZSBM.

Zastosowanie elektronicznej techniki obliczeniowej datuje się od przełomu lat sześćdziesiątych i siedemdziesiątych, z chwilą gdy okazało się, że średnia mechanizacja nie jest w stanie zaspokoić szybko rosnących potrzeb na przetwarzanie danych. Ale jak to bywa zazwyczaj w okresach przejściowych, pierwszy etap wprowadzania informatyki do organizacji i zarządzania cechowały przypadkowość i brak koordynacji. Dopiero Uchwała nr 11 Zarządu CZSBM z 20 lutego 1974 r. (znowelizowana dwa lata później) była próbą unormowania rozwoju zastosowań ETO. Jednakże nie wyeliminowała ona całkowicie działań nieskoordynowanych. O skutkach takich działań nie trzeba się szczegółowo rozwodzić; wystarczy zauważyć, że pociągają one za sobą wzrost nieuzasadnionych kosztów.

Reasumując, należy stwierdzić, że dotychczasowy brak koncepcji jednolitego systemu informatycznego dla CZSBM oraz niedostateczna koordynacja zadań podejmowanych przez spółdzielnie wojewódzkie lub nawet przez poszczególne spółdzielnie mieszkaniowe owocowały w postaci pojawienia się podobnych rozwiązań systemów (np. systemów rozliczeń czynszów), a także rozmaitych koncepcji organizacyjnych i sprzętowych.

Społeczny koszt inwestowania w informatykę w organizacjach spółdzielczych jest szczególnie szybko i bezpośrednio odczuwalny przez członków finansujących wszystkie spółdzielcze przedsięwzięcia. Jest więc oczywiste, że wdrażaniu informatyki do organizacji i zarządzania powinna towarzyszyć rozwaga i rachunek ekonomiczny. Sprowadzać się to powinno do zasady, że ani jedna złotówka czerpana bądź co bądź z kieszeni członka-lokatora (obecnego lub potencjalnego) nie może być wydatkowana lekkomyślnie. Stąd wszelkie projekty zastosowania nowoczesnych rozwiązań w zarządzaniu spółdzielczością mieszkaniową — w tym także projekt zastosowań informatyki — powinny być nie tyle efektowne, nie tyle dostarczać satysfakcji twórcom, co powinny być efektywne i przydatne użytkownikom.

Lektura programu rozwoju informatyki na lata 1978—85 skłania do przeświadczenia, że ta oczywista prawda przyświecała jego autorom.

Pokładanie poważnych nadziei w realizacji nakreślonego na przyszłość programu nie stanowi o negatywnej ocenie wszystkiego, co do tej pory zdrobiono. Fakt, że przed powołaniem koordynatora przedsięwzięć informatycznych na szczeblu Centrali ZSBM spółdzielnie działały w pojedynkę, nie świadczy o tym, że działano w złej wierze.

w spółdzielczości mieszkaniowej

Należy przy tym pamiętać, że specyfika zastosowań informatyki jest w spółdzielczości mieszkaniowej inna niż w każdej innej gałęzi czy branży. Poszczególne spółdzielnie mieszkaniowe są w dużym stopniu autonomiczne wobec spółdzielni wojewódzkiej, te zaś z kolei względem CZSBM. Centralizacja więc jest tu mniejsza niż gdzie indziej, a w ten sposób tradycyjny schemat powiązań organizacyjnych, czyli przepływu oraz agregacji informacji, jest tu trudniejszy do zastosowania.

Obecnie agregacji centralnej podlegają informacje z dwóch sfer działalności spółdzielni: informacje o członkach i kandydatach spółdzielni mieszkaniowych i informacje o realizacji inwestycji. Dane te podlegają emisji do innych resortowych systemów. Pozostałe systemy właściwe dla spółdzielczości mieszkaniowej funkcjonują tylko na użytek wewnętrzny poszczególnych jednostek. Na przykład system gospodarki materiałowej przetwarza informacje interesujące na obecnym etapie rozwoju poszczególne spółdzielnie lub najwyżej spółdzielnie wojewódzkie, podobnie jak system rozliczeń czynszów.

Wymienione tu systemy od dawna pracują na rzecz spółdzielczości, są zresztą wystarczająco efektywne, najwyżej można wytknąć, że te same dziedziny poddane informatyzacji w różnych spółdzielniach były ujmowane w niejednolite systemy — z uwagi na rozmaitość sprzętu lub różne podejścia projektantów.

Z innych, naturalnych względów jeszcze większą rozmaitość trzeba odnotować na obszarze projektowania inwestycji, gdzie komputery zastosowano do obliczeń inżynierskich, kosztorysowania, autokreślenia itp., a projektowanie inwestycji to wcale nie bagatelna sfera działalności spółdzielczości mieszkaniowej. Trzeba bowiem nadmienić, że oprócz zaplecza projektowo-inwestycyjnego przedsiębiorstw budownictwa mieszkaniowego, CZSBM rozwinął własną sieć biur — tzw. INWESTPROJEKTÓW, które przygotowują i projektują rozwiązania dla spółdzielczego budownictwa mieszkaniowego oraz są reprezentantem inwestora wobec wykonawców.

Tak więc automatyzacja spółdzielczości mieszkaniowej obejmuje dwa jej zasadnicze obszary działalności:

- obszar zarządzania, w którego skład wchodzą takie zagadnienia, jak: ewidencja członków i kandydatów spółdzielni, gospodarka materiałowa, rozliczanie rachunków z członkami .
- obszar projektowania inwestycji — projektowanie i kosztorysowanie budynków i osiedli.

Program zakłada, że w pierwszym etapie informatyka będzie stosowana tylko w spółdzielniach mieszkaniowych zrzeszających około 10 tysięcy członków (obecnie istnieje w kraju ok. 45 takich spółdzielni), a w dalszej kolejności w następnych ok. 54 spółdzielniach, które już obecnie osiągają taką liczbę członków. A także we wszystkich spółdzielniach wojewódzkich oraz 14 INWESTPROJEKTACH, przy czym te ostatnie kwalifikowano pod względem wysokości przerobu inwestycyjnego.

W artykule wyeksponowano jako zasadniczą cechę zamierzeń — pragmatyzm. Przejawia się on przede wszystkim w rezygnacji z tworzenia własnej sieci ośrodków obliczeniowych. W okresie formułowania założeń programu rozwoju spółdzielczość mieszkaniowa dysponowała skromnym potencjałem własnego sprzętu.

Na dobrą sprawę nie było żadnego komputera, albowiem ODRA 1304, zainstalowana w 1974 r. w Zakładzie Informatyki Śródmiejskiej Spółdzielni Mieszkaniowej w Warszawie, stanowi własność ELWRO i jest przedmiotem dzierżawy.

Ponadto spółdzielczość ma do swojej dyspozycji kilka minikomputerów MERA (Spółdzielnia Mieszkaniowa na warszawskim Żoliborzu — trzy MERY 305, Specjalistyczny Ośrdek Organizacji i Zarządzania w Łodzi — jedną MERĘ 305, Ośrodek Przetwarzania Danych w Nowej Hucie — dwie MERY 305) oraz 3 minikomputery WANG 2200 (dwa w katowickim i jeden w sopockim INWESTPROJEKCIE).

W dalszym ciągu przetwarzanie informacji na potrzeby spółdzielczości mieszkaniowej będzie zlecane ośrodkom dysponującym wolnymi mocami sprzętu. Takich ośrodków w kraju jest sporo, ale najbardziej dostosowaną do lokalizacji wojewódzkich spółdzielni sieć ma Zjednoczenie Informatyki. W związku z tym program rozwoju odwołuje się do porozumienia zawartego 25 kwietnia 1975 r. między CZSBM a Zjednoczeniem Informatyki. Wydaje się, że skala potrzeb, jakie zarysowują się przed spółdzielczością — z jednej strony — i odpływ szeregu dotychczasowych poważnych klientów ZETO — z drugiej — przyda nowych impulsów dla zadzierzgniętej ongiś współpracy.

Rezygnując z zakupu własnych komputerów, spółdzielnie mieszkaniowe będą mogły przyczynić się do zapełnienia portfela zleceń ośrodków ZI oraz zyskają możliwość organizacji własnych stacji przygotowania danych, a w dalszej kolejności do korzystania z systemów abonenckich w oparciu o własne końcówki skomunikowane z dużymi komputerami ZETO.

Z drugiej strony zaprezentowana koncepcja pozwoli przesunąć punkt ciężkości zainteresowań na prace związane z projektowaniem i wdrażaniem systemów, a nade wszystko z rozpoznawaniem własnych potrzeb w zakresie stosowania automatyzacji. Pozwoli na właściwe koordynowanie przedsięwzięć poszczególnych spółdzielni, prowadząc w efekcie do produkcji powielarnych systemów i oprogramowania. Wyeliminuje to zapewne niekorzystne zjawisko rozpraszania i dublowania prac nad oprogramowaniem.

W dobie rozrachunku z wieloma utopijnymi koncepcjami bombastycznych systemów, program jednolitego systemu informatycznego CZSBM — skromny, realny, odwołujący się do potrzeb, własnych możliwości i możliwości krajowych partnerów — zasługuje na uwagę i pozytywną ocenę. Jest prosty i przejrzysty. Jest zbiorem zaleceń a nie nakazów. Z tym wyjątkiem, że o ile nie skłania żadnej spółdzielni do pochopnej automatyzacji, określa wyraźnie warunki, które winny być respektowane w procesie wdrażania informatyki do organizacji i zarządzania spółdzielczością mieszkaniową.

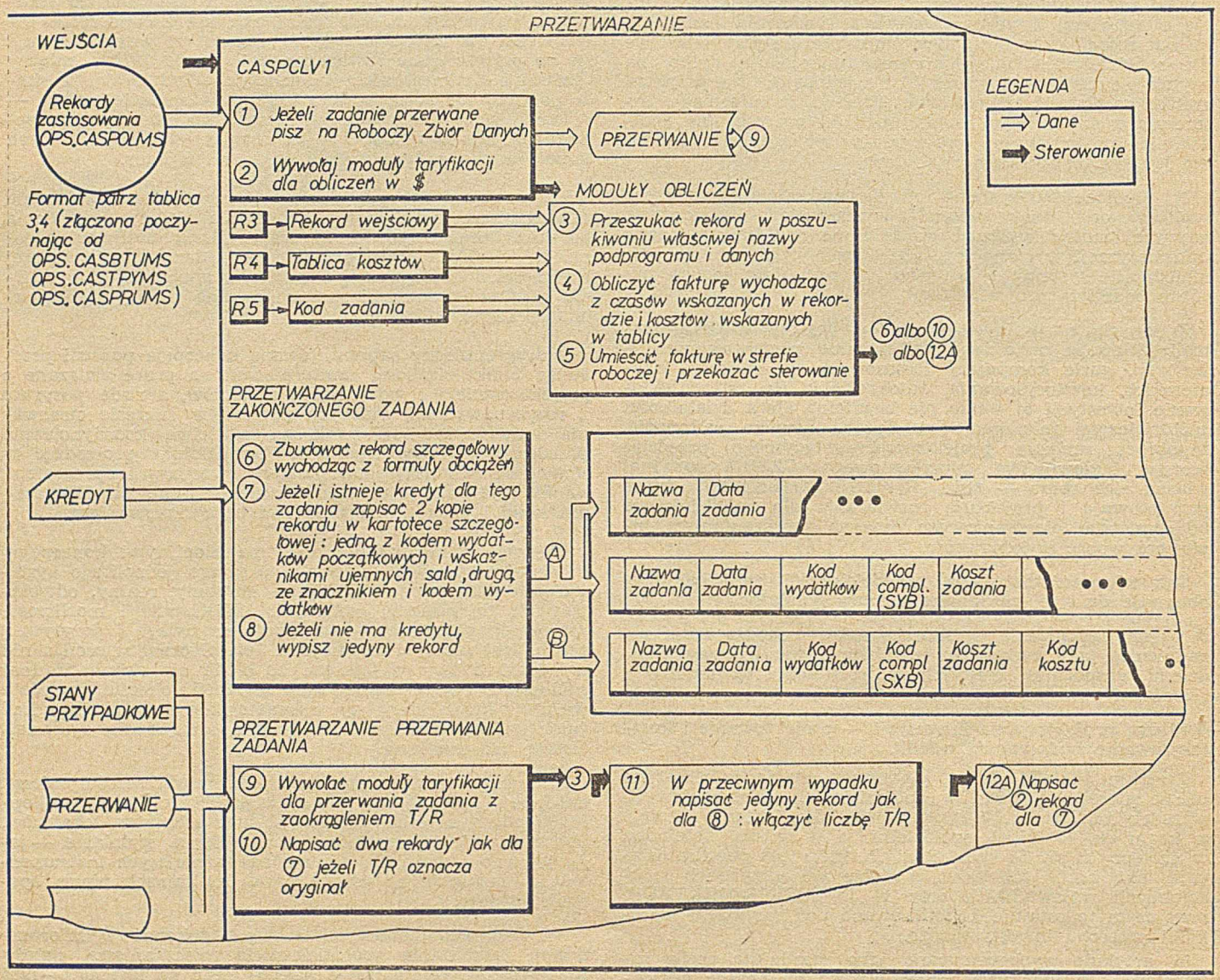
Ponieważ zaś głównym zadaniem spółdzielczości mieszkaniowej jest budowa jak największej liczby jak najlepszych mieszkań oraz sprawna obsługa członków, lokatorów, potraktowanie informatyki we właściwym wymiarze — jako jednego z ważnych, acz nie najważniejszych instrumentów — zasługuje na uwagę nawet na łamach pisma specjalistycznego.

Nie bagatelizując dotychczasowych dokonań w informatyzacji zarządzania spółdzielczością mieszkaniową, należy przypuszczać, że realizacja nakreślonego na lata 1978—1985 programu rozwoju będzie sporym skokiem ilościowym i jakościowym. To, że prolongowano opracowanie koncepcji jednolitego systemu informatycznego CZSBM, może wyjść teraz tylko na dobre. Także dlatego, że w ten sposób można będzie uniknąć błędów, które chyba ongiś byłyby nieuchronne.

Krystyn BERNATOWICZ

Metoda HIPO (2)

Dokończenie artykułu charakteryzującego główne założenia metody projektowania systemów informatycznych HIPO. Bliższe szczegóły na temat tej metody oraz jej praktycznego stosowania zawarte są w wydawnictwie firmy IBM „HIPO — A Design Aid and Documentation Technique", przeznaczonym dla projektantów systemów oraz programistów.



Rys. 7. Schemat szczegółowy

Rysunek 7 przedstawia stosunkowo nieskomplikowany schemat szczegółowy.

Schematy należy kreślić w przewidywanej kolejności hierarchicznej. Ten prosty warunek wydaje się ewidentny dla analityka opracowującego projekt systemu. Staje się on mniej oczywisty, gdy projekt lub też realizacja systemu są zaawansowane i kiedy to próbuje się aktualizować dokumentację.

Fakt zstępowania z góry na dół narzuca pakietowi HIPO strukturę, którą trudno byłoby osiągnąć posuwając się w kierunku przeciwnym. Rozmieszczenie powiązań oraz istniejących relacji wymagałoby wówczas olbrzymiego nakładu intuicji lub zbyt wielkich przybliżeń. W przypadku gdy funkcja systemu została zagubiona w trakcie realizacji, próba zastosowania metody HIPO a posteriori ma wszelkie szanse niepowodzenia.

TABLICA VTOC

Wychodząc z góry, schemat drzewiasty określa niezbędne pakiety funkcjonalne i wskazuje poziom szczegółowości każdego z nich. Ponieważ ten schemat określa zawartość schematów niższego poziomu hierarchicznego, staje się on ogólnym spisem treści systemu zwanym tablicą VTOC. Należy pamiętać o tym, że liczba poziomów schematów jest funkcją złożoności realizowanego systemu. Przy przekroczeniu wartości 4 lub 5, system może stać się zbyt złożony dla jednego podziału HIPO. W tym przypadku można wprowadzić kilka podziałów. Należy obrać taki system numeracji, który będzie wygodny dla tablicy VTOC. Numeracja ta w zasadniczy sposób ułatwia wyszukiwania potrzebnych informacji. Tablica VTOC najczęściej oznaczana jest numerem 0.

Prostokąt umieszczony na szczycie drzewa i odpowiadający schematowi całości systemu będzie identyfikowany symbolem 1. Wewnątrz każdego poziomu schematy mogą być numerowane kolejno np. 2.1, 2.2, 2.3, ... lub można się do nich odwoływać poprzez schemat macierzysty, np.: pierwszy prostokąt trzeciego poziomu znajdujący się pod pierwszym prostokątem drugiego poziomu będzie oznaczony numerem 3.2.1.1., gdzie 3 oznacza trzeci poziom, 2.1. numer prostokąta macierzystego, a 1 — pierwszy prostokąt tego ostatniego. Gdy rozmiary systemu są niewielkie, schematy mogą być numerowane w kolejności poziomej lub logicznej.

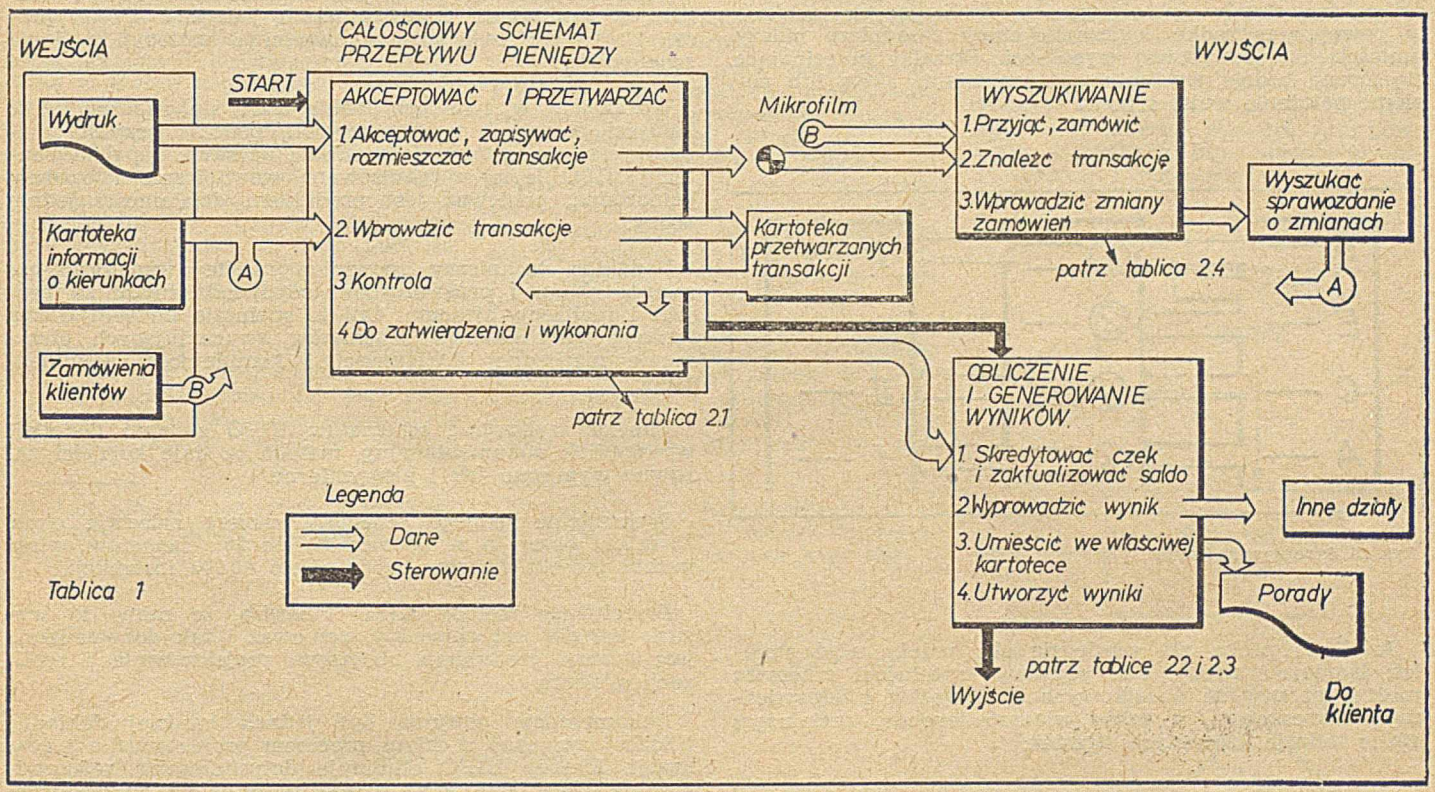
Etapy przetwarzania powinny być również ponumerowane wewnątrz prostokątów opisujących poszczególne funkcje. Ich rozpracowanie odbywa się na bezpośrednio niższym poziomie całości lub na poziomie szczegółowym. Utrzymanie funkcji na stosunkowo ogólnym poziomie schematu całościowego zmniejsza prawdopodobieństwo konieczności dokonywania istotnych zmian w trakcie wykonywania ostatecznej postaci projektu.

Wszystkie schematy odpowiadające pewnemu poziomowi powinny zawierać ten sam stopień szczegółowości. W istocie jedną z podstawowych funkcji metody HIPO jest dostarczanie dokumentacji o sukcesywnie rosnącej szczegółowości. Osoba korzystająca z tej dokumentacji odwołując się do poziomu pośredniego nie powinna napotykać tu ani nadmiaru syntezy ani też nadmiaru szczegółów, natomiast zmierzając w kierunku schematów szczegółowych powinna stwierdzać rosnący stopień złożoności opisu.

Reasumując schematy całościowe odwołują się do schematów bezpośrednio niższego poziomu hierarchicznego, gdzie poszczególne funkcje przedstawione są bardziej szczegółowo.

SCHEMATY SZCZEGÓŁOWE

Opracowanie schematów szczegółowych wymaga na ogół dłuższego czasu, ponieważ w tym stadium wzajemnie przeplatają się etapy projektowania koncepcyjnego



Rys. 8. Schematy funkcjonalne

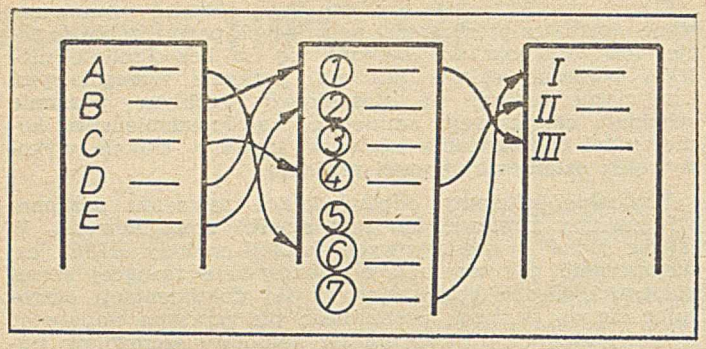
SCHEMATY CAŁOŚCIOWE

Po tablicy VTOC następują schematy całościowe. Ich liczba i zawartość są określone przez rozgałęzienia funkcjonalne tablicy VTOC. W schematach całościowych funkcja jest na ogół wyodrębniona w postaci prostokątów należących do tej części schematu, która opisuje przetwarzanie. Jeżeli schemat jest dobrze przemyślany, to liczba tak wyodrębnionych funkcji odpowiada liczbie schematów funkcjonalnych bezpośrednio niższego poziomu (rys. 8).

Schematy całościowe ukazują jedynie wejścia i wyjścia. W żadnym przypadku nie należy ich opisywać, ponieważ będą one szczegółowo wyjaśnione w schematach najniższego poziomu.

i technicznego. Niemniej jednak przedstawiona poniżej procedura może okazać się użyteczna. Arkusz papieru należy najpierw podzielić pionowo na 3 części, otwarte z dołu. W lewej części należy ponumerować wszystkie możliwe wymagania dotyczące przetwarzania (wejścia) w takim porządku, w jakim przychodzą nam do głowy. W prawej części należy ponumerować w taki sam sposób wszystkie pożądane wyniki (wyjścia). W części środkowej należy ponumerować etapy przetwarzania w takim porządku, w jakim będą one prawdopodobnie wykonywane.

Ostatnim lub raczej przedostatnim etapem wykonywania schematu jest połączenie wejść oraz wyjść z odpowiednimi przetwarzaniami. Otrzymane w ten sposób wyniki będą prawdopodobnie bardzo zagmatwane (rys. 9) i dlatego



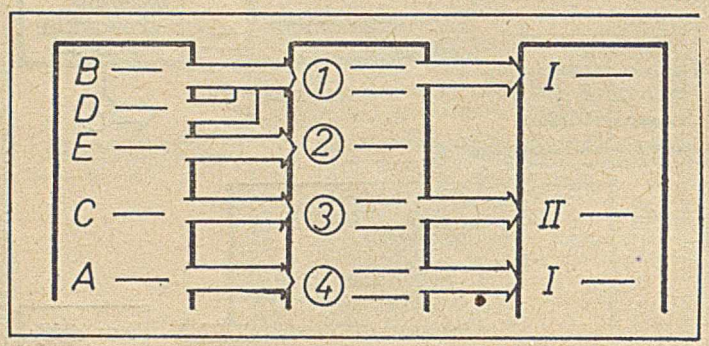
Rys. 9. Przepływ wstępny

należy schemat przerysować wprowadzając następujące zmiany:

— grupować powiązane z sobą wejścia i wyjścia. Używać grubych strzałek przepływu informacji w celu otrzymania obrazu o odpowiedniej skali i proporcji,

— w miarę możliwości łączyć i określać wspólne etapy przetwarzania.

Po dokonaniu tych zmian schemat stanie się bardziej przejrzysty, chociaż wejścia i wyjścia nie będą jeszcze we właściwych relacjach względem przetwarzania. Nie zmieniając etapów przetwarzania realizujących funkcje w pewnym porządku logicznym etapy zasadnicze należy umieścić na jednakowej wysokości. Strzałki przecinające się można będzie wyeliminować zmieniając skalę lub poziome położenie wejść i wyjść (rys. 10).



Rys. 10. Przepływ końcowy

Jeżeli ta próba nie powiedzie się, należy użyć strzałek łączących strony. Dla poprawy czytelności schematu można też używać dodatkowych prostokątów i odpowiednich zakreskowań, a także włączać legendę, definiującą różne rodzaje używanych strzałek.

OPIS ROZWINIĘTY

Po podanym uzupełnieniu schematu można w podobny sposób zredagować odpowiedni opis. Będzie on zawierał po jednej pozycji dla każdego odwołania do procedury schematu. Każda pozycja zawiera opis funkcji, do której się ona odnosi.

Zwięzłość była wprawdzie istotą schematów, ale przy tworzeniu opisów rozwiniętych nie jest wskazana nadmierna oszczędność słów. Aby przetwarzanie nabrało wyrazu, należy dostarczyć całej niezbędnej treści, dotyczącej jego budowy. O ile to możliwe należy utworzyć wzajemne odsyłacze między schematami blokowymi i wydrukami programów. Po zakończeniu schematu i opisu rozwiniętego realizacja może ulegać wprawdzie zmianom, natomiast sam schemat nie powinien wymagać zmian, jeżeli tylko wyeliminowano z niego wszelkie informacje dotyczące realizacji. Przeróbka opisu rozwiniętego jest względnie prosta a uwzględniając potrzebę zawartych w tekście informacji, wysiłek taki jest w pełni opłacalny.

ZALECANY SPOSÓB POSTĘPOWANIA

Jeżeli zamierza się wyciągnąć wszystkie korzyści ze stosowania metody HIPO jako instrumentu projektowania i realizacji, należy równolegle prowadzić zarówno projektowanie, jak i dokumentację systemu. Schematy powinny być rysowane przez ich użytkowników, tzn. projektantów całości i szczegółów systemu lub programistów, ponieważ stanowią one podstawowe narzędzie projektowania i realizacji.

Bardzo pomocny może być główny redaktor techniczny w tym stadium projektu, w którym schematy nabierają ostatecznego kształtu. Redaktor ten może wprowadzić uwagi krytyczne dotyczące schematów, a nawet rozwiązań projektu systemu, jak również narzucić poszczególnym analitykom i projektantom jednolity sposób pracy, wskazując zwłaszcza na właściwe metody kreślenia schematów.

Wprawdzie w tym przypadku traci się znaczną część korzyści płynących z samokontroli, podczas bezpośredniego użycia schematów, ale wzamian uzyskuje się dokumentację HIPO lepiej zorganizowaną wewnętrznie i bardziej jednorodną, ponieważ jest produktem działalności jednej osoby.

Redaktor techniczny powinien ściśle współpracować z analitykami i programistami odpowiedzialnymi za projekt i realizację systemu. Oni to spełniają bowiem ważną funkcję weryfikatorów schematów gwarantujących otrzymanie niezbędnej dokładności i kompletności dokumentacji.

Chociaż wykonanie schematów HIPO stanowi dodatkowy wysiłek dokumentacyjny, metoda ta daje korzyści zarówno wymierne jak i niewymierne.

Wykreślenie jednego schematu zabiera zaledwie około 30 minut pracy, a za pomocą 10 lub 20 schematów opisać można projekt systemu rzędu około 40 000 instrukcji.

Przeciwnicy metody HIPO twierdzą, że mimo to wysiłek ten nie jest opłacalny, ponieważ i tak dokumentacja jest zawsze produktem ubocznym projektowania i realizacji systemu.

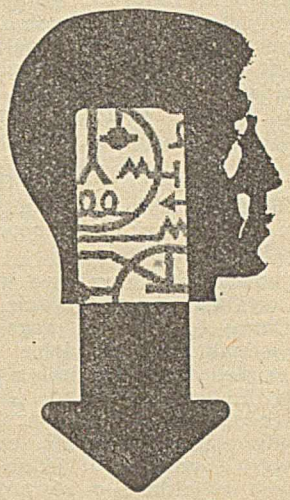
Dokumentacja powinna być jednak bardziej dokładna i ściślej związana z całym procesem projektowania i realizacji. Metoda HIPO eliminuje dotychczasową różnorodność dokumentacji, a w szczególności wiele opisów werbalnych, prowadząc do jej pożądanej zwięzłości i konkretności. Szczególne zalety dokumentowania metodą HIPO uwidaczniają się przy konserwacji systemu oraz szkoleniu jego użytkowników.

Opracowano na podstawie
L'INFORMATIQUE NOUVELLE

Czytelniku!

Pamiętaj o prenumeracie INFORMATYKI

Problemy i decyzje



Aby wygrać partię warcabów, szachów lub GO, komputer, podobnie jak człowiek, musi rozwiązać kilkaset, a może nawet kilka tysięcy problemów. Dlatego właśnie, o czym już wspomniałem, gry z udziałem maszyn cyfrowych były dobrym poligonem doświadczalnym przy tworzeniu sztucznej inteligencji. Ale samo rozwiązywanie problemów jest zagadnieniem dużo ogólniejszym, znacznie przekraczającym doświadczenia opisywane w dwóch poprzednich numerach INFORMATYKI. „W symulowanie przez komputer procesu rozwiązywania problemów włożono chyba więcej pracy niż w symulowanie jakiegokolwiek innego procesu psychicznego" — pisał Michael J. Apter[1]). „Częściowo mógł przyczynić się do tego fakt, że rozwiązywanie problemów wchodzi w skład każdej definicji myślenia, nawet najwęższej".

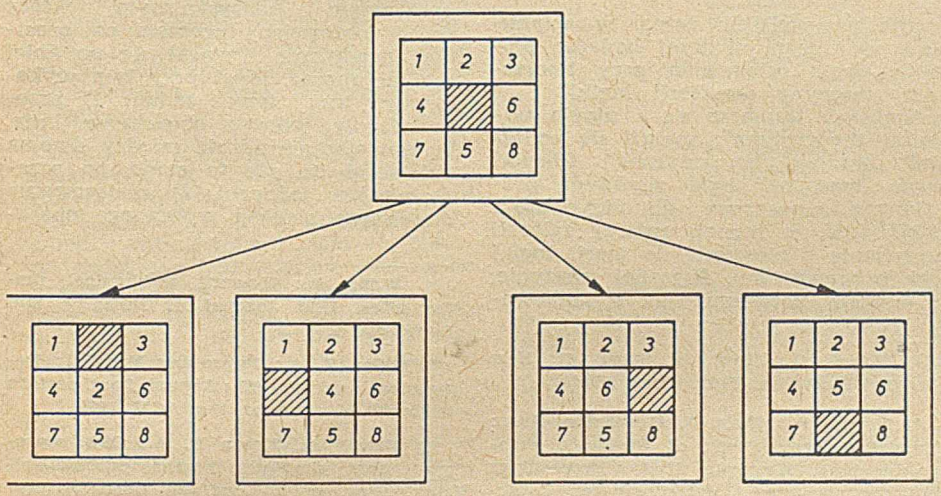
Stając przed problemem, którego nie da się pokonać „z marszu", staramy się najpierw zrozumieć jego istotę i zebrać o nim dostateczną ilość informacji. Potem następuje analiza i czas na „dojrzewanie" problemu, czyli na przetrawienie go, „przespanie się z nim" itp. Efektem owych zabiegów powinno być nawiedzające nas — ponoć ni stąd, ni zowąd — olśnienie i towarzyszący mu okrzyk „eureka", tzw. zjawisko „aha!". I teraz dopiero możemy przystąpić do konkretyzacji rozwiązania — z praktyki zaś wiadomo, że nad tym również trzeba się niemało natrudzić.

Dzieje się tak nawet przy rozwiązywaniu dość prostych łamigłówek, jak choćby ta powszechnie znana o ośmiu ruchomych polach (rys. 1). Problem nie odbiega tu zbytnio od zadań występujących przy grach komputerowych (można stworzyć „drzewo przesuwania", zupełnie analogicznie do „drzewa gry"), a jednocześnie jego złożoność nie wykracza poza przeciętne ludzkie umiejętności.

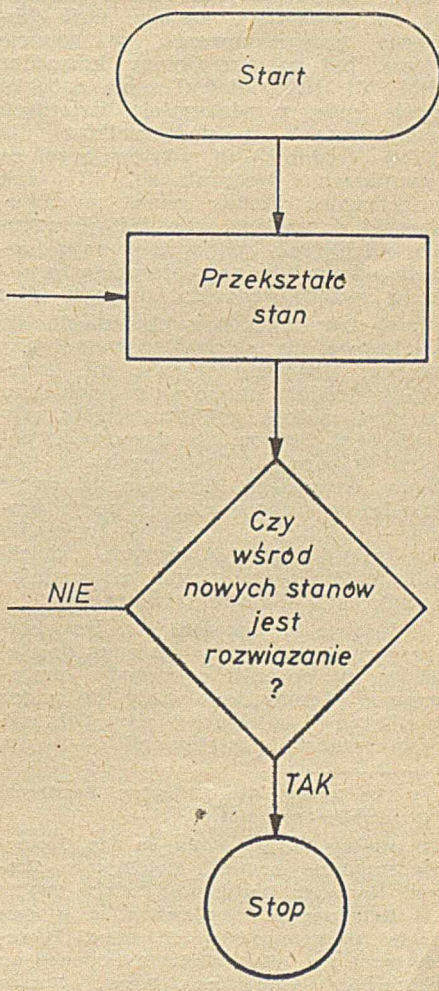
J. E. Doran i D. Michie z myślą o rozwiązywaniu tego typu problemów opracowali na Uniwersytecie w Edynburgu algorytm i program komputerowy nazwany GRAPH TRAVERSER[2]). Problem interpretowany był tutaj jako zbiór stanów i reguł przechodzenia od jednego stanu do drugiego. Algorytm powinien przekształcać kolejne stany i sprawdzać, czy wśród nowo powstałych znajduje się ten, który może być uznany za ewentualne rozwiązanie (rys. 2), lub szukać drogi między stanami przy pewnych warunkach (np. zadanie o misjonarzach i ludożercach).

1) Apter M. J.: Komputery a psychika. Symulacja zachowania. Warszawa, PWN 1973

2) Doran J. E.: New developments of the Graph Traverser. Machine Intelligence 2. Edinburgh, Oliver and Boyd 1968



Rysunek 1



Rysunek 2

GRAPH TRAVERSER został użyty do pokonywania wielu trudności natury praktycznej, których istota zbliżona jest do problemu, łamigłówki ośmiopolowej. W ten sposób m. in. ustalono miejsca postoju ciężarówek w garażu z uwzględnieniem kolejności ich wjazdów i wyjazdów. Ciekawym przykładem użycia GRAPH TRAVERSERA było też układanie znacznie trudniejszej niż poprzednia łamigłówki 15-polowej[3]).

Dużo poważniej niż zabawa w łamigłówki wygląda rozwiązywanie zadań geometrycznych. Oba te zagadnienia nie są jednak zbyt od siebie odległe — ich stopień trudności jest z punktu widzenia komputera niemal identyczny. Przekonał się o tym Marvin Minsky,[4]) gdy chciał sprawdzić, czy maszyna potrafi udowodnić jedno z podstawowych twierdzeń o równości kątów przy podstawie trójkąta równoramiennego. Maszyna zasugerowała rozwiązanie, wcale nie wymagające dodatkowych konstrukcji.

Okazało się, że dowód przeprowadzony przez maszynę został już przedstawiony na przełomie III i IV wieku przez Pappusa z Aleksandrii, ale niestety później zapomniany.

Dowodzeniu twierdzeń z geometrii euklidesowej poświęciła także dużo uwagi grupa badaczy pracująca w nowojorskim centrum doświadczalnym IBM pod kierunkiem H. Gelerntera[5]). Zbudowano tam nawet „maszynę do geometrii", mającą dość obszerne oprogramowanie, w gruncie rzeczy był to przerobiony komputer IBM 704, który dawał sobie całkiem nieźle radę z większością twierdzeń zaczerpniętych z podręczników dla szkoły średniej lub występujących na egzaminach maturalnych. I tak twierdzenie: „Każdy punkt na dwusiecznej dowolnego kąta jest jednakowo odległy od boków tego kąta" zostało udowodnione w niespełna minutę, a przeprowadzenie dowodu dla: „Przekątne równoległoboku dzielą się wzajemnie na połowy" trwało trzy minuty. Powyższe wyniki nie zadowoliły jednak naukowców, zwłaszcza że zbyt złożony program często wyczerpywał będącą do dyspozycji pojemność pamięci. Przez następne lata grupa Gelerntera udoskonalała swoje urządzenie wzbogacając je np. o programy selekcji i programy tworzenia konstrukcji geometrycznych, dzięki którym maszyna mogła udowadniać nawet bardzo złożone twierdzenia.

„Maszyna do geometrii" jest klasycznym przykładem systemu działającego w oparciu o metody heurystyczne. Metody te okazały się nader przydatne w większości prac dotyczących sztucznej inteligencji — np. przy omawianych już komputerowych grach w warcaby, szachy i GO. Różnią się one zdecydowanie od metod algorytmicznych, choć nie należy tych dwóch sposobów postępowania uznawać za skrajnie przeciwstawne. Algorytm to zbiór czynności, których wykonanie doprowadza do rozwiązania, czyli pewnego rodzaju „przepis na wynik". Heurystyka zaś to zestaw ogólnych zasad działania opartych na wynajdywaniu nowych faktów oraz istniejących między nimi zależności. Postępowanie heurystyczne oszczędza czas, bo nie wymaga rozpatrywania wszystkich możliwych wariantów, lecz tylko te, które rokują największe szanse powodzenia. Możemy je zastosować do analizy rozmaitych problemów, nawet takich, dla których nie ma rozsądnego algorytmu. O ile jednak podejście algorytmiczne zawsze gwarantuje rozwiązanie, to heurystyka takiej pewności nie daje. Sugeruje ona raczej propozycje rozwiązań, które zresztą są zazwyczaj całkiem poprawne.

Kariera heurystyki rozpoczęła się na dobre w 1954 r., dzięki teoretycznej pracy prof. G. Polya[6]). Stało się wtedy rzeczą oczywistą, że heurystyczny sposób myślenia jest znacznie bliższy naturze działania naszego mózgu niż rozumowanie algorytmiczne. „Heurystyka — pisali E. A. Feigenbaum i J. Feldman — to oparta na doświadczeniu zasada, strategia, chwyt, uproszczenie lub jakiś inny środek, w sposób istotny ograniczający poszukiwanie rozwiązań złożonych problemów".

Propagatorzy heurystyki, A. Newell, J. C. Shaw i H. Simon, wykazywali jej przewagę posługując się argumentem „Algorytmu Muzeum Brytyjskiego" (nazwanym swego czasu przez polskiego tłumacza „brytyjskim algorytmem muzealnym"!). Chodziło o to, że gdyby rozdać małpom maszyny do pisania i nauczyć algorytmów posługiwania się tym sprzętem, to kiedyś wystukałyby one wszystko, co zawierają książki zgromadzone w British Museum. Tyle że kosztowałoby to mnóstwo czasu, nie mówiąc już o papierze. Lepiej więc stosować heurystykę.

Nie przypadkiem zatem w artykule[7]), w którym ci trzej badacze opisują swoje osiągnięcia przy tworzeniu „maszyny do teorii logiki" i w przypisach tłumaczą się z użycia terminu „heurystyka", znalazł się podtytuł „studium heurystyczne". Chociaż owa „maszyna" była właściwie programem napisanym dla komputera JOHNIAC, to jednak autorzy zalecali „myślenie o maszynie do teorii logiki jak o prawdziwej, fizycznej maszynie, i o pracy programu, jak o zachowaniu się maszyny". Programowanie heurystyczne umożliwiło JOHNIAKOWI śledzenie tylko tych łańcuchów logicznych, które prowadziły do uzyskania poprawnego dowodu, przy czym kolejne kroki były wykonywane z wystarczająco dużym prawdopodobieństwem. W ten sposób komputer wzniósł się ponad dotychczasowy poziom mechanicznego rozwiązywania problemów, a jego praca nabrała subtelności i mogła świadczyć o posiadaniu czegoś w rodzaju intuicji.

„Maszyna" została użyta do dowodzenia twierdzeń logiki formalnej (rachunku zdań) w oparciu o zadane aksjomaty i reguły wnioskowania. Swoje umiejętności potwierdziła dowodząc 38 z 52 twierdzeń umieszczonych w drugim rozdziale książki B. Russela i A. N. Whiteheada „Principia Mathematica". Z twierdzeniem: „(p implikuje nie p) implikuje nie p" uporała się w 10 sekund, a „nie (p lub q) implikuje nie p" zajęło jej 12 minut. Przy dowodzeniu twierdzenia: „[p lub (q lub r)] implikuje [(p lub q) lub r] dała jednak za wygraną po 23 minutach.

Napisanie programu, zdolnego udowodnić wszystkie zawarte w „Principia Mathematica" twierdzenia (choć nie heurystycznego), udało się w trzy lata później H. Wangowi[8]). Zastosował on maszynę IBM 704, a metody dowodzenia oparł na specjalnych formalnych regułach wnioskowania. W 1963 r. J. R. Slagle opracował program, który sprostał 52 spośród 54 zadań wziętych z egzaminów końcowych na M.I.T. Jeszcze bardziej udaną wersję całkowania formalnego stworzył w 1967 r. J. Moses. Trzeba tu również wspomnieć o kilku wariantach programu do układania twierdzeń opartych na zadanych aksjomatach — powstały one w 1967 r. na Uniwersytecie Berkeley — oraz o zdolnym do samouczenia się programie analizy regresyjnej, napisanym na M.I.T.[9])

Kontynuacją „maszyny do teorii logiki" był opracowany przez tych samych autorów[10]) program GPS (General Problem Solver — Rozwiązywacz Problemu Ogólnego). GPS był programem heurystycznym, który dzielił postawiony problem na prostsze zagadnienia i starał się po kolei z nimi uporać. Mógł przy tym wykonywać trzy typy zadań: 1) przekształcanie jednego obiektu w inny, 2) zmniejszanie różnic między dwoma obiektami (to nie to samo, co przekształcanie jednego), 3) zastosowanie do danego obiektu wybranego opera-

3) The Robots Are Comming. Sandback, NCC Publications 1974

4) Minsky M. L.: Notes on the Geometry Problem. Artificial Intelligence Project. Hanover: Dartmouth College 1956

5) Gelernter H.: Realization of a geometry theorem-proving machine. Proceedings of the International Conference on Information Processing. Paryż, Unesco House 1959

6) Polya G.: How to Solve It. New Jersey, Pincerton Univ. Press 1954

7) Newell A., Shaw J. C., Simon N. A.: Empirical explorations of the logic theory machine. Heuristic Study. Proceedings of the Western Joint Computer Conference 1957

8) Wang H.: Toward mechanical mathematics. IBM Journal of Research and Development 1960

9) Miller F. A.: Improving Heuristic Regression Analysis. 6th Annual Southeastern Regional Meeting of the ACM 1967

10) Newell A., Simon H. A.: GPS, a program that simulates human problem-solving. Proc. of a Conf. on Learning Automata. Karlsruhe 1961

tora. Po każdej operacji GPS sprawdzał, czy nowo powstały stan jest trudniejszy do rozwiązania od poprzedniego. Jeśli tak było, rezygnował z tego kierunku poszukiwań, wybierając prostszą drogę.

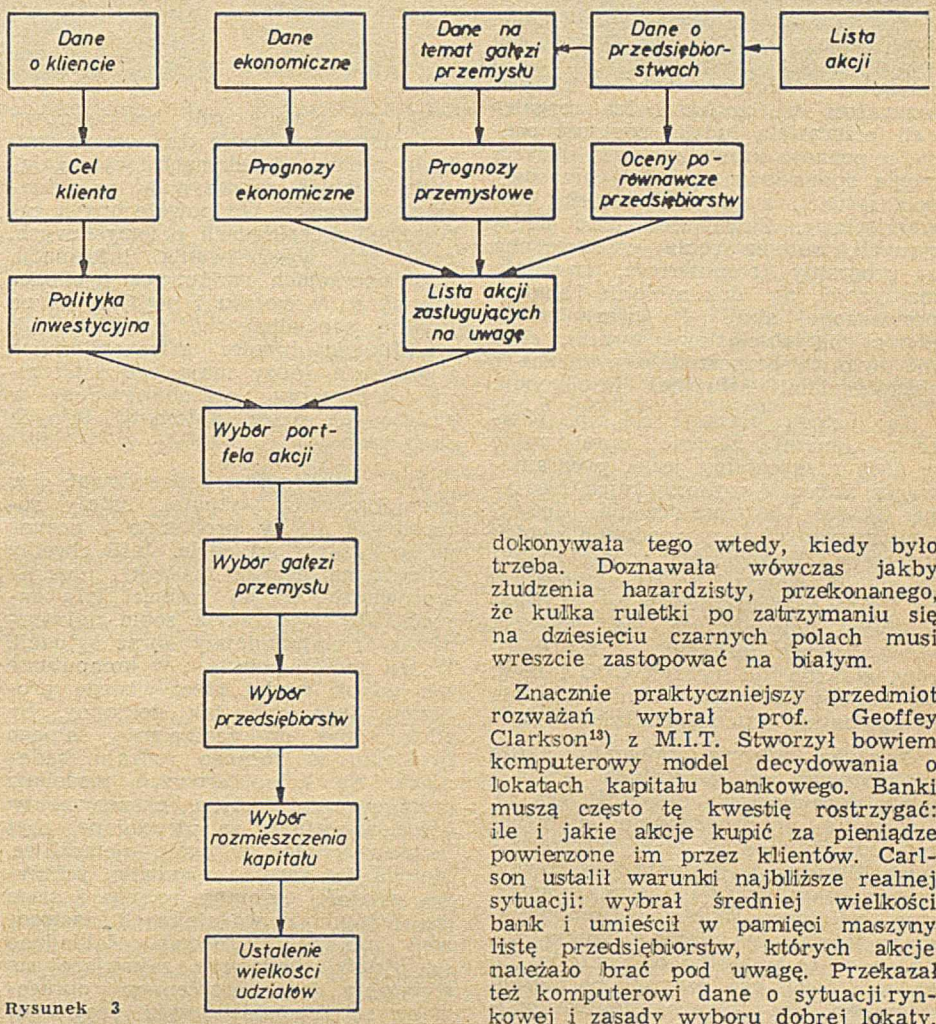
Parę dokładnych zapisów działania GPS-u porównano z protokołami zawierającymi relacje osób nakłonionych do rozwiązania tych samych problemów. Okazało się wtedy, że nie tylko generalna metoda (wyodrębnianie podproblemów, przekształcanie opisów i stosowanie znanych prawideł), ale i niektóre szczegółowe decyzje maszyny nie odbiegały od toku myślenia przeciętnego człowieka. GPS stał się więc nie tylko doniosłym osiągnięciem w dziedzinie sztucznej inteligencji, ale przysłużył się także lepszemu poznaniu ludzkich procesów myślowych. Komputerowe programy rozwiązywania problemów — zdaniem Newella i Simona — „dowodzą z absolutną pewnością, że swobodne zachowanie inteligentnego człowieka można uznać za wynik działania skomplikowanego, lecz skończonego i w pełni określonego zbioru reguł".

Oprócz posługiwania się metodą podziału na podproblemy, czyli rozwiązywania postawionego zagadnienia krok po kroku, GPS miał możliwość działania w inny sposób. Mógł upraszczać złożone zadanie, rozwiązywać je w owej mniej skomplikowanej postaci, a rezultaty wykorzystywać przy rozwiązywaniu właściwego problemu. Te dwie drogi były zresztą zazwyczaj stosowane w późniejszych eksperymentach nad automatycznym rozwiązywaniem problemów.

Oczywisty sprawdzian zdolności komputerów uzyskano egzaminując je przy pomocy typowych testów na inteligencję. Takich, w których dla określenia spostrzegawczości, umiejętności analizy, syntezy, dokonywania wyboru, oceny itp., kojarzy się słowa, np. w pary „mało—dużo", „źle—dobrze" lub zestawia figury poszukując prawidłowości wynikających z poprzednich układów. Badań takich przeprowadzono bardzo wiele, najbardziej znane wykonane zostały przez Thomasa Evansa. Program Evansa nazwany „Analogy"[11]) i przeznaczony do rozwiązywania testów geometrycznych przez wyszukiwanie analogii między figurami, zniechęcał jednak użytkowników swoją nadmierną złożonością.

Dalsze prace rozproszyły wysiłki naukowców w różnych dziedzinach, którym potrzebne były urządzenia rozwiązujące ich konkretne problemy. Wiele z tych dokonań stało się zresztą podstawą późniejszych sukcesów osiąganych przy komputerowym rozstrzyganiu zagadnień ekonomicznych, technicznych i militarnych. Specyfika tych dyscyplin spowodowała jednak, że trzeba było odstąpić od ściśle zdeterminowanych sytuacji i zająć się przypadkami bliższymi rzeczywistości. Należało brać pod uwagę elementy, które nie zawsze dadzą się przewidzieć, ale mogą mieć istotny wpływ na rezultaty. Zaczęto więc zastanawiać się nad zmianami powstającymi w procesie rozwiązywania problemu przez maszynę w wyniku pojawienia się warunków niepewności. Komputer zdolny do podejmowania decyzji, których następstwa nie są w pełni znane, mógłby okazać się znacznie bardziej przydatny niż „rozwiązywacze problemów" działające w ściśle i do końca określonych warunkach.

Najprostszy proces decyzyjny, czyli wybór jednej z dwu możliwości (tak lub nie, czarne lub białe) — tzw. wybór binarny — odtworzył w maszynie cyfrowej Julian Feldman.[12]).



Rysunek 3

Program przewidywał następne zdarzenie (np. to, czy w kolejnym momencie lampka będzie zgaszona czy zapalona) w oparciu o hipotezy wysnute na podstawie obserwacji poprzednich sytuacji. Zbiór owych hipotez składa się zatem, powiedzmy, z przepisów: „skoro było jasno, to powinno być ciemno", „skoro było jasno dwa razy pod rząd, to powinno być dwa razy ciemno", „po jednym wygaszeniu następują zwykle dwa zapalenia". Do tego zbioru dochodził zbiór hipotez przeciwnych, tj. „po zapaleniu powinno być zapalenie" itd. — na wypadek, gdyby włączający i wyłączający światło zorientował się, że maszyna wykryła jego system i zaczął działać na opak. Wtedy i maszyna przejdzie na system odwrotny stosując owe hipotezy przeciwne. Modyfikacja hipotez pozwoliła komputerowi na decydowanie z trafnością bardzo niewiele odbiegającą od ludzkich poczynań. Sporo kłopotu sprawił jednak moment przechodzenia na hipotezy zanegowane — maszyna nie zawsze dokonywała tego wtedy, kiedy było trzeba. Doznawała wówczas jakby złudzenia hazardzisty, przekonanego, że kulka ruletki po zatrzymaniu się na dziesięciu czarnych polach musi wreszcie zastopować na białym.

Znacznie praktyczniejszy przedmiot rozważań wybrał prof. Geoffey Clarkson[13]) z M.I.T. Stworzył bowiem komputerowy model decydowania o lokatach kapitału bankowego. Banki muszą często tę kwestię rostrzygać: ile i jakie akcje kupić za pieniądze powierzone im przez klientów. Carlson ustalił warunki najbliższe realnej sytuacji: wybrał średniej wielkości bank i umieścił w pamięci maszyny listę przedsiębiorstw, których akcje należało brać pod uwagę. Przekazał też komputerowi dane o sytuacji rynkowej i zasady wyboru dobrej lokaty, przekazując mu schemat procesu podejmowania decyzji przez finansistów (rys. 4).

Tak przygotowana maszyna zmierzyła się z urzędnikiem bankowym, którego wieloletni staż obdarzył niemal podświadomą umiejętnością korzystnego lokowania kapitału.

11) Evans T.: A Heuristic Program to Solve Geometric Analogy Problems. Proc. AFIPS. Ann. Spring Joint Computer Conference 1964

12) Feldman J.: Simulation of behaviour in the binary choice experiment. Proc. of the Western Joint Computer Conference 1961

13) Clarkson G. P. E.: Portfolio Selection: A Simulation of Trust Investment. Englewood Cliffs 1962

Podczas pierwszej próby urzędnik i komputer uznali, że najrozsądniej będzie rozdzielić fundusze między pięć przedsiębiorstw. Cztery typowali podobnie, a przy ocenie wysokości lokat zgadzali się w dwóch pozycjach. Druga próba wypadła lepiej. Człowiek i maszyna wybrali dziewięć firm, w tym siedem tych samych i tylko raz ustalili inne sumy wkładu. Próby trzecia i czwarta dały wręcz niewiarygodne wyniki. Ilość i wartość lokat okazały się w obu wypadkach identyczne, a rozbieżność wystąpiła na jednej tylko pozycji — co do wyboru przedsiębiorstwa.

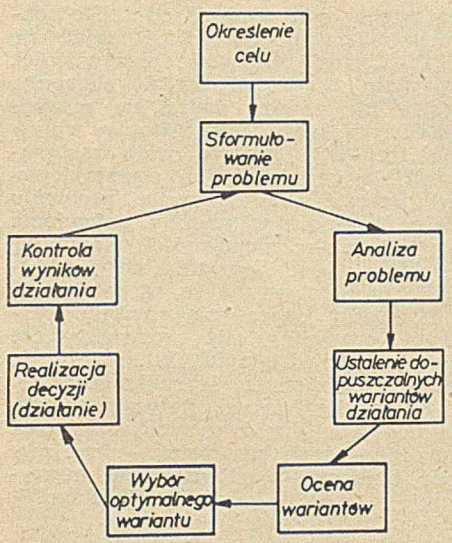
Kilka amerykańskich instytucji zainicjowało w latach sześćdziesiątych stosowanie tzw. tablic decyzyjnych, które zawierały macierzowe zestawienia zasad podejmowania decyzji. Tablice podzielone na parę obszarów i wykorzystujące prostą implikację (JEŚLI — TO) zdawały egzamin w wielu sytuacjach, zbyt złożonych, by nadawały się do rozstrzygnięcia z pomocą zdrowego rozsądku. W przypadkach, w których rozmaitość wariantów i ich rozgałęzień przerastała ludzką zdolność oceny, a ponadto każda następna decyzja mogła spowodować, że dotychczasowy opis sytuacji stanie się całkiem bezwartościowy. Zawartość tablic decyzyjnych, automatycznie przetworzona na programy komputerowe, pozwalała maszynom na dokonywanie analizy prowadzonej akcji i gwarantowała niemal natychmiastowe wyniki, choć nie było to bezpośrednio związane z zagadnieniami sztucznej inteligencji.

Tą metodą posłużył się np. Koncern General Electric opracowując w 1960 r., specjalny język programowania Tabsol i wykorzystując maszynę GE-200 do sprawdzenia decyzji podejmowanych przy produkcji swoich wyrobów. W dwa lata później odbyło się w Nowym Jorku sympozjum na temat komputerowych tablic decyzyjnych. Pojawiło się też wówczas wiele nowych języków i translatorów, głównie dla maszyn CDC i IBM. Do Europy ta nowa technika dotarła na początku lat siedemdziesiątych.[14])

Maszyna cyfrowa wykazała swoją przydatność na wszystkich etapach cyklu podejmowania decyzji[15]) (rys. 5). Udział jej w decydowaniu o wyborze jednego spośród wielu dopuszczalnych kierunków działania jest równie ważny co w decyzjach precyzujących sam cel tego działania. Komputery, rzecz jasna, niekoniecznie muszą odpowiadać za wszystko od początku do końca. Mogą być jedynie partnerem człowieka albo tylko asystentem podpowiadającym korzystne rozwiązania, czy też zwracającym uwagę na jakiś problem lub okoliczność.

14) Niedźwiedzki J.: Logiczne tablice decyzyjne. Problemy Informatyki. Warszawa. Ośrodek Badawczo-Rozwojowy Informatyki 1975

15) Proces decyzyjny. Europejski Program Badawczy Diebolda. Warszawa, OBRI 1973 nr 28



Rysunek 4

W tej właśnie roli komputerowy konsultant występuje dziś we wszystkich niemal dziedzinach wymagających szybkich rozstrzygnięć w sytuacjach złożonych. Od powszechnych zastosowań w zakładach przemysłowych, systemach wyszukiwania informacji, przy operacjach bankowych i ekonomicznych, w wojsku i polityce, aż po wąskie specjalistyczne przypadki, jak weryfikacja hipotez astronomicznych i ustalanie, kiedy mają być zniszczone dokumenty przechowywane w strzeżonych przez automaty tajnych archiwach.

Aby właściwie zadecydować w skomplikowanej sytuacji, należy zorientować się w problemie i poznać go w każdym szczególe. Jeśli chcemy polegać na decyzji maszyny, musimy wprowadzić do jej pamięci tyle danych, aby wytworzyła sobie najwierniejszy i najpełniejszy obraz sytuacji. W ten sposób powstaje w komputerze jak gdyby model rzeczywistego problemu, jego opis tak dokładny, by zdolny był do zastąpienia samego problemu w różnego rodzaju badaniach. Na komputerowych modelach przeprowadzać można eksperymenty, które w naturze są utrudnione, zbyt kosztowne, albo w ogóle niemożliwe. Metoda ta, zwana symulacją, przeżywa ostatnio ogromny rozwój i staje się jednym z ważniejszych zastosowań maszyn cyfrowych. Zakładane są ośrodki badawcze zajmujące się wyłącznie tym problemem, opracowane są specjalne języki programowania (GPSS, SIMULA, SIMSCRIPT, CSL).

Modele komputerowe umożliwiają spojrzenie z lotu ptaka na duże i pogmatwane zagadnienia, a jednocześnie pozwalają skoncentrować się na którymś z wybranych fragmentów dla podejmowania decyzji z dowolnego szczebla i w rozmaitych aspektach. Matematycy ze Stanford Research Instytute, Evans, Wallace i Sutherland, proponują np. system wyszukiwania informacji o wydarzeniach na świecie oraz symulację komputerową decyzji podejmowanych w polityce wewnętrznej i zagranicznej[16]). Na wypadek, gdyby nie były one zbyt trafne, autorzy zamieszczają w swej książce model gry synchronicznej na mapach z uwzględnieniem różnych rodzajów wojsk i szczebli dowodzenia, od dywizji do plutonu. Rzecz rozpatrywana jest z wielu punktów widzenia. W symulacji procesu walki można znaleźć model ułatwiający decyzje w czasie bezpośredniego starcia, ocenę elementów ochrony terenowej i ukrycia, widoczności, rozpoznania, celności strzału i szacowanie strat. Jest też symulacja radarowej obserwacji pocisku i przebiegu ewakuowania rannych. Taki model pozwala na przerzucenie w nerwowej sytuacji wielu czynności dowódczych na maszynę — ludziom pozostają funkcje wykonawcze, a więc walka.

Symulacją innego rodzaju zmagań zajęli się czescy informatycy. Niecierpliwiąc się przed poprzednimi mistrzostwami świata w piłce nożnej, przekazali komputerowi wiosną 1974 r. komplet informacji o uczestnikach rozgrywek. W odpowiedzi maszyna zadecydowała, że zwycięstwo przypadnie drużynie RFN i zdania roztropnie nie zmieniła, mimo że testowano ją paroma różnymi programami. „Wyliczenia te — powątpiewał korespondent „Trybuny Ludu" — polscy sympatycy futbolu potraktują w myśl znanego powiedzenia „Poczekamy, zobaczymy, piłka jest okrągła..."

Nawet jeśli komputerowe werdykty i nie zawsze dorównywać będą ludzkim, to przecież skłaniają one człowieka do krytycznej oceny i doskonalenia własnych zdolności decyzyjnych. Sformułował to trafnie prof. Peter Druckner z Nowego Jorku: „Komputer zmusił nas do uporządkowania samych siebie. Jest on największym formalistą, jakiego udało mi się kiedykolwiek widzieć, i na tym polega jego największy wpływ na proces podejmowania decyzji. Zmusza nas do uprzedniego przemyślenia naszych założeń i powoduje, że zdajemy sobie w ogóle sprawę z tego, że przyjmujemy jakieś założenia".

16) Evans G. W., Wallace G. F., Sutherland G. L.: Symulacja na maszynach cyfrowych. Warszawa WNT 1973

Marek HOŁYŃSKI

SEIK – system ewidencji informacji

Użytkownicy (od lewej): Paweł Odyniec, Karol Pawełczyk i Tadeusz Czyrkiewicz

Projektanci (od prawej): Bogdan Fiutowski, Tadeusz Kuna, Grażyna Zienkiewicz

Kolejnym systemem OBRI, prezentowanym na naszych łamach (patrz także INFORMATYKA nr 2, 4/78), jest SEIK, znajdujący efektywne zastosowanie w różnych dziedzinach zarządzania przedsiębiorstwem. W początkowej fazie projektowania prace prowadzone były pod hasłem „kadry", ale ostateczny kształt systemu pozwala na wiele innych zastosowań.

Efektem wstępnych założeń w pracach projektowych są łatwość i niskie koszty jego modyfikacji. Jest to cenna zaleta w przypadkach konieczności wprowadzenia dużej ilości zmian i częstej aktualizacji zbiorów.

Powiązanie SEIK z pakietem programowym JAZ-75 pozwala uzyskać możliwość dodatkowego rozszerzenia zakresu zastosowań.

SEIK może być realizowany na komputerze ODRA 1304, ale lepsze efekty uzyskać można stosując komputer ODRA 1305. Wymagany minimalny zestaw dla maszyn ODRA serii 1300 to: 4 jednostki pamięci taśmowej, pamięć operacyjna 32 K oraz czytnik kart i drukarka wierszowa. Aby wdrożyć system trzeba przeprowadzić w przedsiębiorstwie wstępne prace organizacyjne.

O tych problemach oraz wielu innych, które pojawiły się w fazie projektowania a później w przedsiębiorstwach eksploatujących system, rozmawiali projektanci i przedstawiciele użytkowników. W rozmowie udział wzięli: Bogdan Fiutowski, kierownik Zakładu Oprogramowania SOIK OBRI, Tadeusz Kuna, główny specjalista—Zakładu Oprogramowania SOIK OBRI, Grażyna Zienkiewicz, konsultant w Zakładzie Obrotu Oprogramowaniem OBRI, Andrzej Aranowski, kierownik Zakładu Obrotu Oprogramowaniem OBRI, Paweł Odyniec, kierownik Pracowni EPD w Fabryce Przekładni Samochodowych POLMO w Tczewie, Tadeusz Czyrkiewicz, programista Pracowni FPS POLMO w Tczewie, Karol Pawełczyk, projektant Zakładów Przemysłu Meblarskiego w Radomsku, oraz Andrzej Klimek, redaktor INFORMATYKI.

INFORMATYKA: Zanim zaczniemy mówić o produkcie, porozmawiajmy o jego założeniach programowych i technice rozwiązania.

T. Kuna: Pierwsze prace podjęliśmy w roku 1972, kiedy mogliśmy już korzystać z doświadczeń nagromadzonych podczas wielu prac eksploatacyjnych różnych systemów. Rzeczywistość stale zmienia się, a wraz z nią muszą zmieniać się metody i techniki zarządzania. System, który obejmuje pewne jego dziedziny, musi mieć możliwość dostosowywania się do zmian w koncepcji zarządzania, musi być odpowiednio elastyczny. Jedno z możliwych rozwiązań spełniających ten warunek to jest rozwiązanie parametryczne, czyli pewien abstrakcyjny model, pozwalający realizować stosunkowo szeroki zakres funkcji, przy uwzględnieniu oczywiście możliwości sprzętowych i techniczno-organizacyjnych. Ale parametryzacja może być przyjęta zarówno bardzo wąsko, jak i bardzo szeroko. Sprawa odpowiedniego doboru jej poziomu jest bardzo istotna, ponieważ zbyt mała parametryzacja nie rozwiązuje problemu, natomiast nadmierna, szczególnie na wolnych maszynach, powoduje nieefektywność działania.

Sparametryzowaliśmy przede wszystkim budowę struktury zbiorów, wejście i związane z nim procedury kontroli, czyli zakładanie i utrzymywanie zbiorów (aktualizacja i modyfikacja). Jest praktyczna możliwość definiowania dowolnego dokumentu występującego w trakcie przetwarzania. Oczywiście jego wielkość została ograniczona ze względu na wymagania techniki oprogramowania, ale praktycznie jest bardzo duża.

INFORMATYKA: Jakie informacje można otrzymać z systemu?

T. Kuna: Zanim określę dwa podstawowe typy informacji, które można uzyskać, chciałbym przypomnieć, że w SEIKU otrzymujemy je stawiając pytanie w pewnym języku algorytmicznym, sformalizowanym zgodnie z obecnie stosowaną techniką (języków zupełnie swobodnych kompilować jeszcze nie potrafimy).

Pierwszy typ informacji uzyskujemy podając warunek wyboru, który wskazuje pozycje do umieszczenia w tabulogramie wynikowym. Język pozwala na dokonanie według żądanego klucza selekcji dokumentów, określenie podzbioru dokumentów oraz uporządkowanie ich na wydruku. Dla realizacji selekcji wybraliśmy formuły boolowskie — formy matematyczne jednoznaczne i najbardziej rozpow-

szechnione. Istnieją w języku odpowiednie zwroty pozwalające wybierać odpowiednie pozycje ze zbioru oraz klucze dla określania sortowania. Można zadeklarować, że zapytanie będzie dotyczyło pewnego zbioru, czyli nie musi odnosić się do ustalonego stałego zbioru.

Druga forma informacji, możliwej do uzyskania w SEIKU, związana jest z opisem tablic. Oznacza to, że użytkownik stawiając pytanie musi zdefiniować tablice, które pozwolą mu na uzyskanie informacji w postaci zestawieniowej. Do tego celu stosowany jest również pewien język. Tablica miałaby oczywiście najbardziej ogólną formę gdyby każdy element opisany był pewną formułą. Wymaga to jednak bardzo wiele pracy, nawet przy niezbyt dużych wymiarach tabeli. W praktyce wystarczy jednak podanie formuł dla wierszy i kolumn, a wtedy liczba ich wzrasta liniowo w stosunku do wymiarów tablicy. Dla określenia budowy wierszy i klumn również wykorzystano formuły boolowskie.

W naliczaniu tablic przyjęto dwie alternatywne możliwości. Można opisując warunek dla kolumny i wiersza zażądać zliczenia liczności (np. w „kadrach"), albo też na przecięciu tych elementów policzenia sumy pewnej pozycji z dokumentu (np. „środki trwałe").

W tablicy występują elementy, które są naliczane bezpośrednio ze zbioru — kolumny i wiersze. Istnieje także możliwość definiowania pewnych kolumn i wierszy w wyniku działań arytmetycznych na tych kolumnach i wierszach. A zatem jest możliwość konstruowania wieloszczeblowej agregacji danych. Dopuszcza się trzy poziomy takiej agregacji. Przyjęliśmy pewien standard w redagowaniu tabulogramu. Zrezygnowaliśmy z pełnej swobody w określaniu tablicy, która prowadziłaby do konieczności perforowania kilkuset znaków na kartach. Standard dotyczy spraw formalnych, takich jak redagowanie nagłówka, rysowanie główek, szpaltowanie, obliczanie szerokości tablic. Jak dotychczas standard ten odpowiadał użytkownikom. Nowych możliwości w tym zakresie dostarcza system JAZ.

INFORMATYKA: W jakich sferach organizacji przedsiębiorstwa SEIK może być najlepiej wykorzystany?

T. Kuna: Żeby podjąć jakąś pracę trzeba powiedzieć, do czego produkt będzie przeznaczony. Taki warunek stawia życie. Przyjęliśmy, że będzie to system kadrowy, chociaż myśleliśmy nie tylko o kadrach.

INFORMATYKA: Podał Pan interesujące fakty dotyczące możliwości zastosowań, ważne będą także informacje o funkcjach systemu, pozwalających na przekształcanie informacji.

T. Kuna: Sprawa przekształcenia, czyli naliczania pewnych informacji dzięki wprowadzeniu innych informacji, jest sprawą modyfikacji zbiorów. Trzeba było skonstruować pewien język, który pozwolił na odpowiednio łatwe programowanie. Ze względu na duże tempo prac przyjęliśmy rozwiązanie najprostsze, w którym na przykład zmienną roboczą trzeba definiować w zbiorze. Język ten pozwala przetwarzać zbiory ewidencji. Każda instrukcja jest instrukcją warunkową, czynności są ponumerowane i wykonywane po spełnieniu określonego warunku. Jest to pewien odpowiednik tablic decyzyjnych.

Przekształcenia oferowane w SEIKU wykorzystują w pełni przyjętą parametryzację — to bardzo ważny atrybut systemu. Procedury bazują na automatyzmie parametryzacji, a nie na jego konkretnej reprezentacji. Oznacza to, że wykorzystany jest w pełni fakt sparametryzowania zbiorów.

Program reorganizacji daje możliwość przeorganizowania zbioru według określonej nowej struktury, bez konieczności ponownego zakładania, uzupełniając go potrzebnymi informacjami.

Tworząc język musieliśmy zastosować pewne uproszczenia, które pozwoliły wtedy (sześć lat temu) zrealizować podjęte prace. Polegały one na wprowadzeniu różnych „formalizmów", jak seperatory, kropki przed i po operatorze, itp. Nie mogliśmy pójść na pełną swobodę.

Kilka słów o technikach programowania. System jest zrealizowany jako interpreter. Pytanie lub tablica jest przekształcana przez program kompilujący na pewną tablicę, która powstaje w pamięci maszyny w języku maszynowym dla niej wygodnym. Powstała struktura jest następnie interpretowana, to znaczy, wszystkie programy są typu interpretacyjnego.

Najnowszym osiągnięciem w dziedzinie oprogramowania jest oprogramowanie strukturalne. Programując kilka lat temu, nie stosowaliśmy jeszcze tych technik, ale oprogramowanie zrobiliśmy już modularnie.

INFORMATYKA: Czy były jakieś nieoczekiwane trudności podczas uruchamiania systemu?

T. Kuna: Jestem zwolennikiem zasady, że należy dobrze przygotowywać programy i w konsekwencji rzadziej wchodzić na maszynę, a jeżeli się już wchodzi, to robić dużo, nie improwizować. Dlatego zastosowałem prosty sposób, który na początku wymagał wprawdzie wykonania pewnej dodatkowej pracy, a w efekcie pozwolił mi na bardzo szybkie uruchomienie programu.

Pod każdą etykietą występującą w programie wstawiłem komentarz, który na tabulogramie daje mi wydruk tej etykiety i wartości kluczowych zmiennych. Po przeanalizowaniu w otrzymanych tabulogramach około 2 tysięcy etykiet wiedziałem już jak program poszedł, nic już nie musiałem przypuszczać, nic nie musiałem zgadywać! Było to bardzo duże przyspieszenie w uruchomieniu programu. Jak wiadomo takie śledzenie drogi programu stosowane jest już we wszystkich współczesnych językach.

Najwięcej problemów mieliśmy z programami kompilacyjnymi. Wynikały one z braku doświadczenia w robieniu kompilatorów. Były to nasze pierwsze prace tego typu. Zostały opracowane struktury wewnętrzne, pozwalające na przekompilowywanie z języka zewnętrznego, tzn. zdefiniowanie bazy projektowej. Natomiast w realizacji założyliśmy samodzielność w ramach zdefiniowanego fragmentu systemu, to znaczy poszczególne osoby realizowały wyodrębnione fragmenty logiczne.

Dość kłopotliwą sprawą było także obliczenie arytmetycznych formuł boolowskich. Zastosowaliśmy tak zwany zmienny przecinek. Uważaliśmy, że gdy założymy stałe wielkości, to trzeba przewidzieć maksymalne. Mogą wtedy wystąpić nadmiary. Aby zachować dokładność maksymalną — przy 12 cyfrach — zastosowaliśmy własny zmienny przecinek. Można było również oprzeć się na zmiennym przecinku ICL-owskim, co skróciłoby nieco prace. Dokładność została osiągnięta, ale w praktyce okazało się, że tego typu liczby występują bardzo rzadko.

B. Fiutowski: SEIK ma wiele cennych zalet. Jedną z najważniejszych jest pełna parametryzacja wejścia, możliwość deklarowania różnych ewidencji, a także użycia różnych maszynowych nośników informacji. Użytkownik może wprowadzać dodatkowe nośniki informacji, które powstają w związku z wprowadzanymi nowymi dokumentami. Stosunkowo liczny jest także asortyment wydawnictw, które można otrzymać. Ponadto istnieje praktyczna możliwość pisania programów w bardzo dogodnym języku.

Użytkownik systemu SEIK dostaje narzędzie, które umożliwia mu magazynowanie tego co chce, otrzymywanie wydawnictw w formie jaka mu odpowiada oraz deklarowanie własnych algorytmów. Ze względu na tę ostatnią własność system pozwala na różnorakie zastosowania. Bo gdyby był rzeczywiście robiony z myślą tylko o kadrach, to wystarczyłyby funkcje przechowywania danych i tworzenia potrzebnych wydawnictw, nic więcej. Natomiast w innych zastosowaniach, na przykład dotyczących środków trwałych, istotne są algorytmy i możliwość przekształceń ewidencji. Uważaliśmy, że możliwość dowolnego wejścia, duża elastyczność na wyjściu i przekształcania, to jest właściwie wszystko, co można dać do dyspozycji użytkownikowi.

INFORMATYKA: Zalety, o których usłyszeliśmy od Panów pozwalają przypuszczać, że system jest rzeczywiście uniwersalnym narzędziem w zakresie różnych zastosowań. Bywa często, że wraz ze zwiększeniem możliwości użytkowych, ze wzrastaniem

liczby zastosowań, pojawiają się duże trudności w korzystaniu z narzędzia, powodując, że tylko stosunkowo mały krąg osób może sobie poradzić z jego wykorzystaniem. Co mogą powiedzieć użytkownicy o eksploatacji tego teoretycznie doskonałego produktu?

P. Odyniec: W FPS w Tczewie eksploatujemy SEIK od 1975 roku. Naszym zdaniem jest on bardzo dobry i przydatny dla każdego użytkownika. Są jednak pewne problemy występujące przy wdrażaniu. Największym z nich jest zbyt skąpa dokumentacja udostępniana przez OBRI. Ta dokumentacja wystarcza podczas normalnej eksploatacji, ale my podjęliśmy budowę drugiego modułu, przeznaczonego dla środków trwałych (oprócz wykorzystywanego dla kadr). Należało samodzielnie rozwiązać wiele problemów, ale potrzebna jest w takim przypadku szczegółowa dokumentacja.

Wadą SEIK jest zbyt obszerna redakcja w pewnych programach. Na przykład przy aktualizacji słowników niepotrzebny jest ich cały wydruk. Wprowadzenie tylko jednej pozycji do słownika powoduje jego cały wydruk, co jest oczywiście zbędne. A można byłoby wydrukować tylko tę pozycję, która uległa aktualizacji. Tak samo jest przy ankietach.

Chciałbym zwrócić uwagę tych, którzy mają zamiar wdrażać ten system, że SEIK pracuje najlepiej na ODRZE 1305. W niektórych przypadkach, na przykład podczas realizacji tablic, czas pracy skraca się dziesięciokrotnie w porównaniu do ODRY 1304.

Mamy taką jedną tablicę, której wykonanie na ODRZE 1304 trwa 3 godziny, natomiast na ODRZE 1305 zaledwie kwadrans.

T. Czyrkiewicz: Wydaje się, że podczas przetwarzania jest za dużo komunikatów na monitorze, co powoduje zbędne zatrzymania. Gdyby użytkownik miał zestaw tych komunikatów, to w każdej chwili mógłby podawać operatorowi, co ma robić.

K. Pawełczyk: Ja także uważam, że dla aktualizacji niepotrzebny jest wydruk całych słowników. W naszym przedsiębiorstwie, w ZPM w Radomsku, mamy założoną ewidencję kadrową dla około 4 tysięcy pracowników.

W przypadku tak dużych zbiorów trudno jest sprawdzać aktualizację, bo wydruk trwa około 8 godzin, nawet na ODRZE 1305. Do tablic nie ma zastrzeżeń. Natomiast niepotrzebne są procenty z dokładnością do 5 miejsc po przecinku, ponieważ zmniejsza to czytelność wydawnictwa.

Bardzo dobrą cechą systemu jest deklarowanie pól z powtórzeniami i bez powtórzeń. Pola z powtórzeniami mogą odzwierciedlać poprzednie stany.

Uważam, że instrukcja wydana przez OBRI jest mało czytelna, że można z jej pomocą redagować tylko nieskomplikowane tablice. Natomiast skomplikowane operacje, takie jak przenoszenie do pewnych pól z powierzchniami, wymagają konsultacji OBRI. Poza tym w instrukcji jest za mało przykładów. Mało jest również przykładów ze schematem weryfikacji.

Nasz system SEIK został zastosowany do przedmiotów nietrwałych. Jest to pierwsze, pionierskie zastosowanie. Spotkaliśmy się tu z ciekawym problemem, że nie można było pomnożyć ceny przez stopę umorzenia. Na przykład, gdy mnożyliśmy cenę przez stopę umorzenia 0.20, to przy przenoszeniu do wartości przychodu, jeżeli w wyniku była liczba całkowita, to system przenosił wartość o 40 groszy mniejszą. W tej sytuacji cenę pomnożyłem przez 2, a następnie podzieliłem przez 10 i dopiero wtedy otrzymałem prawidłową wartość.

B. Fiutowski: Zgadzam się z tym, że w starej dokumentacji przykłady są nadmiernie uproszczone i jest ich niewiele. Kiedy ona powstawała, proste były również zastosowania. Słuszny jest również zarzut o rozrzutności w wydrukach kontrolnych. Ale są już wprowadzone pewne zmiany. Można sterować wydrukiem za pomocą daty, żądać wydruku z dnia dzisiejszego lub spełniającego określone warunki datowe. Uwzględniono także wprowadzenie dysków magnetycznych jako nośników informacji. Kiedy projektowaliśmy, sądziliśmy, że należy zmusić użytkownika do tego, ażeby zawsze miał aktualny wydruk słownika.

Jednak obecnie, ze względów oszczędnościowych, należało odejść od tej koncepcji.

W sprawie „przeredagowania" programów mogę powiedzieć tak: projektując i programując zrobiliśmy założenie, że użytkownik przy maszynie nie będzie potrzebował żadnej dokumentacji. Muszą istnieć zatem pewne sygnały o rozgałęzieniach programu (programy są wielofunkcyjne) i musi następować konwersacyjne „dogadanie się" z systemem. A to jest opłacone zmniejszeniem szybkości pracy.

W nowej wersji uwzględniliśmy podawanie informacji za pomocą karty. Instrukcje o tym, jaką drogą program ma przejść w danej chwili, nie są podawane.

Spotkaliśmy się także z uwagą dotyczącą stosowania instrukcji, a mianowicie, że operator może się łatwo pomylić, ponieważ taśma przed sortowaniem i po sortowaniu ma taką samą nazwę. Pomyłka taka jest możliwa tylko wtedy, gdy operator jest nieuważny. Jednakże w praktyce takie pomyłki zdarzały się, a ponieważ mogą one nawet powodować zniszczenie ewidencji podczas aktualizacji, zatem jest to uwaga dość istotna.

P. Odyniec: Największym problemem użytkownika, który realizuje pierwsze tablice, jest diagnostyka, to znaczy określenie, jakie mają one błędy. Przy wykorzystaniu otrzymanych opisów błędów pojawiały się bardzo duże trudności z określeniem, co oznacza dany komentarz do błędu i jakich błędów można się spodziewać. W dokumentacji powinno być opisane, że określony komentarz, otrzymany w wyniku niespełnienia jakiegoś warunku, pozwala spodziewać się takiego lub innego błędu.

Jest informacja w dokumentacji, że można ponumerować karty w tablicy, co pozwalałoby postawić diagnozę, która karta jest błędna. Próbowaliśmy stosować numerowanie, ale nie potrafiliśmy tego zrobić. Ważnym problemem dla początkującego użytkownika jest ustalenie miejsca, w którym powstał błąd podczas definiowania tablicy.

Przy wdrożeniu SEIK dla środków trwałych powstał jeszcze inny problem. Po realizacji tablicy schematu weryfikacji zachodzi konieczność ustalenia zgodności z księgowością. Ale w tym schemacie wartość numeryczna podawana jest w zapisie ósemkowym. Tego księgowość odczytać nie może.

Po trzyletniej eksploatacji systemu kadrowego liczba informacji w kartotece stała się zbyt duża. Mamy obecnie około 80 danych o każdym pracowniku i chcielibyśmy część z tych informacji wyeliminować. Niestety nie otrzymaliśmy potrzebnego do tej operacji programu REOR.

B. Fiutowski: Program REOR jest udostępniany każdemu użytkownikowi, który do nas się o to zwróci. Natomiast został on świadomie usunięty z biblioteki, ponieważ mieliśmy pewne kłopoty z pierwszymi użytkownikami. Polegały one na tym, że wykonywali oni reorganizacje, które w wielu przypadkach kasowały w zbiorach informacje potrzebne użytkownikowi. Dlatego wprowadziliśmy następujące utrudnienie. Gdy program ten jest potrzebny, pojawia się sygnał na pulpicie operatora, że nie ma go w bibliotece. Wtedy użytkownik telefonuje do nas i pyta dlaczego nie otrzymał tego programu. Po wyjaśnieniu, które informacje znikną w wyniku zastosowania tego programu, przesyłamy go użytkownikowi.

Problem diagnostyków w definicjach tablic i pytaniach jest trudny do rozwiązania. W większości przypadków ważniejsze jest doświadczenie w deklarowaniu pytań i tablic oraz w lokalizacji błędów niż opisy. Większość błędów powstaje jako efekt błędów wcześniejszych. W wielu przypadkach błąd nie jest sygnalizowany, ponieważ został już jakoś zinterpretowany. Chcielibyśmy, aby program ten został poprawiony, ale jego autor jest obecnie dyrektorem Departamentu Informatyki i nie wiemy, czy będzie mógł to wykonać.

INFORMATYKA: Wiemy, że pod hasłem „kadry" powstał system, który może być stosowany w wielu dziedzinach. Były już udane próby różnych zastosowań i są nadal podejmowane nowe. A komu OBRI najbardziej poleca ten system?

G. Zienkiewicz: SEIK był dotychczas polecany do wszystkich zastosowań ewidencyjnych. A więc ewidencja maszyn i urządzeń, środków trwałych, prostych płac, zamówień, rozliczania czasu pracy maszyn itp., a zatem nie tylko kadr. W górnictwie, na przykład, został zastosowany do ewidencji złóż węgla brunatnego. W takich i podobnych zastosowaniach, których było dotychczas około 40, wykorzystuje się najbardziej zalety tego systemu. W tradycyjnie konstruowanych systemach zorientowanych dziedzinowo użytkownik napotykał na wiele problemów, których obecnie może uniknąć. Wynikały one z tego, że ustalone z góry struktury, zakresy, rekordy, dokumenty wejścia, w przypadku dokonywania zmian były bardzo pracochłonne i wiązały się z dużymi kosztami. Wykorzystując system SEIK można uniknąć tworzenia specjalnych dokumentów, wykorzystując już istniejące w przedsiębiorstwie, a każdą zmianę można wprowadzić podczas eksploatacji w ciągu kilku godzin.

Pytań technicznych dotyczących użytkowania jest niewiele, natomiast w początkowym okresie powstają bardzo istotne problemy związane z ustaleniem relacji pomiędzy samym produktem i rzeczywistością, w której jest eksploatowany.

Najważniejszym chyba zadaniem jest opracowanie systemu tak, aby system działał sprawnie i dawał największe korzyści. Pod tym względem użytkownik jest wprawdzie twórcą swojego systemu, ale bardzo często liczy na pomoc OBRI. Jednak doświadczeni fachowcy z tego ośrodka, którzy mogą łatwo porównać organizację danego przedsiębiorstwa z wieloma innymi eksploatującymi już system, mogą udzielić cennych rad w okresie wdrażania.

INFORMATYKA: Przedsiębiorstwo, które chce wdrożyć SEIK, musi spełniać pewne warunki, musi dysponować odpowiednim sprzętem. Jak powinno być ono przygotowane do podjęcia rozmów o zakupie systemu. Założenia projektowe były robione z myślą o pewnej klasie użytkowników. Jakie warunki należy spełnić?

G. Zienkiewicz: Poza komputerem ODRA, najlepiej 1305, w przedsiębiorstwie powinien istnieć zespół informatyków, który zna produkt, a jednocześnie wie, na czym polega zagadnienie wprowadzenia tego systemu w jego przedsiębiorstwie i jakie zadania mają być przez system spełnione. Potrzebny jest pewien zakres wiedzy, który umożliwi samodzielne utworzenie tablic, a zatem potrzebna jest określona wiedza fachowa.

A. Aranowski: Chciałbym podkreślić, że mamy do czynienia z produktem programowym, a nie systemem użytkowym do konkretnego zastosowania. Jest oczywiste, że przekazujemy go profesjonalistom, a więc nie może to być zarzutem. Użytkownikowi gwarantujemy potrzebne konsultacje. Nie możemy natomiast poprzez nieograniczone konsultacje tworzyć systemu użytkowego. W kilku przypadkach udzieliliśmy użytkownikom konsultacji znacznie szerszych, aby zdobyć doświadczenia w niektórych przypadkach zastosowań. Zebrane materiały były podstawą do uzupełnienia dokumentacji, która w pierwszej wersji miała istotne braki. W drugiej wersji podaliśmy przykłady zastosowań, zebrane już podczas tych konsultacji. Tak więc prezentujemy zastosowania dla środków trwałych, przedmiotów nietrwałych, zamówień, kadr i płac. Ta forma dokumentacji powinna chyba zadowolić przeciętnego kontrahenta. Brakuje jednak takiego „podręcznika użytkownika", który nie będąc dokumentacją użytkową w sensie eksploatacyjnym, byłby przewodnikiem użytkowania w poszczególnych dziedzinach organizacji przedsiębiorstwa. Chcemy, aby dostarczając taki podręcznik dla organizatora, umożliwić użytkownikowi zastosowanie systemu w taki sposób, który nie narzuca mu robienia dodatkowych formularzy i ogranicza zmiany tylko do rzeczywiście niezbędnych. Dokumentacja natomiast powinna służyć profesjonalistom informatykom. W określonych dziedzinach powinny istnieć „metodyki zastosowania", tzn. dokumentacja organizacyjna systemu.

SEIK jest już dość długo eksploatowany. Jest wielu użytkowników, którzy doskonale znają ten produkt.

Do nich właśnie chcemy zwrócić się o napisanie „metodyk zastosowania" w poszczególnych dziedzinach. Będzie to bardzo duża pomoc dla tych, którzy zamierzają wykorzystać ten system. Uważam, że ta dyskusja redakcyjna udowodniła, że produkt żyje, że jest ceniony, chociaż ma pewne mankamenty, z których zdajemy sobie sprawę.

P. Odyniec: Chciałbym podzielić się z przyszłym użytkownikiem praktyczną uwagą. W zakładzie musi być osoba znająca dobrze programowanie, która umie wykonywać schematy logiczne tablic, schematy weryfikacyjne i pytania. Jedna osoba ze znajomością programowania może zupełnie wystarczyć do wdrożenia określonej dziedziny. Gdyby natomiast zamiast wykorzystania SEIK ktoś chciał wykonać system całkowicie samodzielnie, a więc opracować założenia, projekt techniczny i oprogramowanie, wówczas zadanie takie wymagałoby pracy czterech lub pięciu osób w ciągu około 2 lat, powodując znaczne koszty, a jednocześnie małą elastyczność systemu. SEIK daje szczególnie dużą dowolność budowania tablic statystycznych i statystyczno-decyzyjnych. Do tej pory ludzie w zakładzie nie chcą mi wierzyć, że mogę szybko zbudować dowolną tablicę.

W naszej branży motoryzacyjnej jest jednoznaczna tendencja do instalowania komputerów IBM i Jednolitego Systemu. Nasze Zjednoczenie będzie miało swój system kadrowy. Kiedy przy okazji jednego ze spotkań pochwaliłem się, że mam już działający system kadrowy, który daje mi możliwość budowania dowolnej tablicy, otrzymałem natychmiastową reprymendę: „możesz swój „supersystem" wyrzucić, bo na pewno otrzymasz komputer serii RIAD". Myślę, że w takiej sytuacji będę musiał mieć dwa systemy, bo nie mogę zrezygnować z tego, który całkowicie zaspokaja moje potrzeby.

Chciałbym zapytać, czy można podjąć prace nad taką modyfikacją systemu SEIK, aby mógł być on eksploatowany na komputerach Jednolitego Systemu. Myślę, że byłoby wielu nabywców. Uważam także, że wskazane byłoby również zorganizowanie spotkania użytkowników tego systemu. Rozszerzyłoby to możliwości zastosowań w przedsiębiorstwach już eksploatujących SEIKA.

A. Aranowski: Zagadnienie przeprogramowania na Jednolity System jest problemem technologicznym. Rozwiązania można się spodziewać chyba w niedługim czasie, wtedy gdy opanujemy technikę emulacji programów. Istnieje jednak kwestia optymalizacji rozwiązań, bo przy emulacji na ogół powstają straty w postaci zwiększenia czasów przebiegów. Myślę jednak, że w niedługim czasie będziemy mogli zaproponować system SEIK dostosowany do przetwarzania na RIADZIE.

W tym roku przewidujemy dwa spotkania użytkowników, co ułatwi nam prace nad konserwacją systemu. W perspektywie wydaje się, że wskazane byłoby założenie klubu użytkowników systemu SEIK, który umożliwiłby wymianę doświadczeń i uwag pomiędzy ośrodkami eksploatującymi i zespołem autorskim.

Inną inicjatywą OBRI jest projektowane wydawanie biuletynu użytkownika, w którym można byłoby sygnalizować o wszelkich sprawach związanych z systemem SEIK.

G. Zienkiewicz: Chciałabym przedstawić pewną propozycję OBRI — nową ofertę dla użytkowników. W najbliższym czasie każdy użytkownik otrzyma nową, ulepszoną wersję systemu. Zmieniona została technologia przetwarzania, nie będzie w niej zbędnych komunikatów na konsoli (pulpicie operatora). Wprowadzono wspomaganie dyskowe, co znacznie przyspieszy realizację obliczeń. Ponadto zmieniono strukturę fizyczną rekordów, dotychczas zapisywały się bloki po 100 słów, natomiast w nowym ujęciu, po przekształceniu wynikowym, zapis jest w blokach po 512 słów, czyli niekiedy cały logiczny rekord. Umożliwia to współpracę SEIKA z innym produktem OBRI, którym jest Język Adaptacyjny Zapytań — JAZ. Rozwiązanie takie rozszerza możliwości wydawnicze, a szczególnie pozwala na dowolne budowanie tablic.

Opracowanie tekstu i zdjęcia:
Andrzej KLIMEK

Zgodnie z zapowiedzią zamieszczamy zwięzłą charakterystykę następnego, jednego z wyjątkowo atrakcyjnych produktów programowych Ośrodka Badawczo-Rozwojowego Informatyki. Jest to pakiet programowy JAZ 75, który może zainteresować szczególnie szeroki krąg potencjalnych użytkowników.

Pakiet programowy JAZ 75

Pakiet programowy JAZ 75 jest kompletnym oprogramowaniem języka funkcjonalnego o tej samej nazwie.

Cechą wyróżniającą języki funkcjonalne wśród języków programowania jest ich ukierunkowanie na funkcje realizowane przez komputer.

Języki funkcjonalne pojawiły się przed kilku laty niemal równolegle z pojęciem bazy i banku danych. Programy pisane w tych językach składają się w zasadzie z szeregu werbalnych instrukcji, tj. kluczowych słów, zbliżonych często do języka naturalnego, uzupełnionych parametrami stosownie do konkretnego problemu.

Język adaptacyjny zapytań został ukierunkowany na funkcję przetwarzania komputerowego określoną jako „wykonywanie obliczeń i wyprowadzanie wyników". Posiada niezbędne cechy właściwe językom funkcjonalnym. W koncepcji języka znalazły odzwierciedlenie uogólnienia dotychczasowych doświadczeń w dziedzinie przetwarzania danych.

Pakiet programowy JAZ 75 opracowany został dla komputerów ODRA serii 1300 o następującej minimalnej konfiguracji:

— pamięć operacyjna o pojemności 32 K słów
— monitor operatora
— czytnik kart dziurkowanych
— dziurkarka taśmy papierowej
— drukarka wierszowa
— dwie jednostki pamięci taśmowej.

Pakiet programowy JAZ 75 może być zastosowany dla każdego systemu informatycznego spełniającego następujące wymagania:

1) Zbiory danych znajdują się na taśmach magnetycznych, zapisanych pod kontrolą SAP

2) Dane o rekordach występują w postaci:

- liczby binarnej krótkiej
- liczby binarnej podwójnej precyzji
- liczby zmiennoprzecinkowej
- ciągu znaków (maksymalnie 128)
- daty binarnej w standardzie ISO

3) Dane potrzebne do realizacji jednego pytania mogą występować co najwyżej w pięciu różnych zbiorach.

JĘZYK ADAPTACYJNY ZAPYTAŃ UMOŻLIWIA:

— operowanie danymi przy minimalnych wiadomościach o wewnętrznej organizacji ich zbiorów

— krótkie i zwarte formułowanie pytań komunikatywnych dla użytkownika

— elastyczność i uniwersalność w ramach problemu.

Poza tym JAZ redukuje średnio prawie 80 do 90 procent czasu programowania i uruchamiania programów wydawniczych. W podobnym stopniu zmniejsza również czas komputera zużyty na testowanie programów.

Język adaptacyjny zapytań pozwala dostatecznie szybko realizować zmieniające się w czasie wymagania użytkownika w stosunku do systemu informatycznego, bez dodatkowych nakładów pracy wyspecjalizowanych programistów.

Przykładem udogodnień, jakie stwarza pakiet JAZ 75, jest prównanie czynności oraz wykonawców niezbędnych do uzyskania tej samej informacji przy zastosowaniu pakietu DMS (Data Management Software — firmy ICL):

DMS	JAZ 75
1. Sformułowanie zagadnienia dotyczącego sprawozdania (użytkownik)	1. Sformułowanie zagadnienia dotyczącego sprawozdania na arkuszu programowym JAZ 75 (przeszkolony użytkownik)
2. Konwersja zagadnienia na język parametrów DMS i ewentualnie wstawka w języku PLAN (programista w DMS i PLAN)	2. Tłumaczenie programu przez translator JAZ 75 (komputer)
3. Generowanie programu źródłowego w języku PLAN (komputer)	3. Przebieg programu wynikowego — uzyskanie sprawozdania (komputer)
4. Kompilacja i konsolidacja programu (komputer)	
5. Przebieg programu wynikowego — uzyskane sprawozdanie (komputer)	

Podstawowym celem języka JAZ 75 jest maksymalne uproszczenie procesu programowania.

Język adaptacyjny zapytań pozwala ograniczyć programowanie do zdefiniowania wyników przetwarzania.

Programowanie w JAZIE ogranicza się do:

— formułowania pytań maksymalnie zbliżonych do naturalnego formułowania tego typu problemów

— zapisywania pytań na standardowych dla języka formularzach, co ułatwia pracę oraz eliminuje możliwość powstawania błędów.

Prostotę programowania język JAZ uzyskuje dzięki wąskiej specjalizacji i ukierunkowaniu na konkretny system informatyczny. Ukierunkowanie polega na zdolności adaptacji języka do potrzeb określonego systemu. W procesie tłumaczenia przez komputer pytania źródłowego użyte w tym pytaniu identyfikatory danych stają się identyfikatorami własnymi (bazowymi) języka.

Pakiet JAZ 75 pozwala na przetwarzanie zbiorów taśmowych „poziomo", w zakresie jednego wiersza wyników, jak i „pionowo" (sumowanie kolumn wyników w określonej liczbie cykli). Pozwala realizować żądania użytkownika, dotyczące pojedynczych informacji i dowolnie dużych zestawień.

Warunki sprzedaży produktów OBRI

W stosunkach z nabywcami wszystkich produktów systemowych OBRI (systemów, pakietów programowych lub programów) obowiązują Standardowe Warunki Techniczne Sprzedaży (SWTS), które są częścią składową zawieranych umów sprzedaży. Istnieje możliwość sprzedaży również w innym trybie niż podany w SWTS.

Zgodnie z SWTS przedmiotem sprzedaży mogą być:

- produkt systemowy
- dokumentacja techniczna
- usługi OBRI, obejmujące

— instalację u nabywcy lub wskazanego przez niego użytkownika

— szkolenie personelu nabywcy lub wskazanego użytkownika w zakresie eksploatacji produktu.

Terminy realiazcji sprzedaży produktu, dokumentacji i świadczonych usług określone są każdorazowo w umowie kupna/sprzedaży.

FORMA I WARUNKI DOSTAW

Produkty systemowe OBRI dostarczane są nabywcy lub wskazanemu przez niego użytkownikowi w postaci wynikowej, zapisanej na maszynowych nośnikach informacji.

Nabywca lub użytkownik obowiązany jest dostarczyć do OBRI swoje własne nośniki informacji w dobrym stanie technicznym. OBRI rezerwuje sobie prawo do oceny stanu technicznego przekazywanych maszynowych nośników informacji. Jest to zastrzeżenie konieczne dla zapewnienia odpowiedniej jakości dostarczonych produktów programowych.

Typ, ilość oraz termin dostawy maszynowych nośników informacji określany jest w umowie kupna/sprzedaży.

Dokumentacja techniczna produktu systemowego przekazywana jest nabywcy/użytkownikowi w trzech kompletach egzemplarzy.

WARUNKI INSTALACJI PRODUKTU

Przez instalację produktu należy rozumieć przystosowanie produktu do jego eksploatacji na zestawach komputerowych wymienionych w umowie kupna/sprzedaży. Przystosowanie do eksploatacji obejmuje wprowadzenie do systemu operacyjnego komputera niezbędnego oprogramowania umożliwiającego użytkowanie produktu zgodnie z jego opisem użytkowo-funkcjonalnym.

System operacyjny oraz zestaw komputerowy, na którym dokonuje się instalacji produktu, muszą spełniać określone wymogi techniczno-funkcjonalne.

Ze względu na to, że OBRI bierze pełną odpowiedzialność za sprawność swego produktu, musiano zachować dla siebie prawo oceny spełnienia ww. wymogów, a w przypadku zastrzeżeń zachować dla siebie prawo do odstąpienia od instalacji produktu.

Instalacja produktu dokonywana jest przez wyspecjalizowany zespół serwisowy OBRI. Nabywca obowiązany jest zapewnić bezpłatnie OBRI określoną w umowie kupna/sprzedaży liczbę godzin pracy zestawu komputerowego wraz z zabezpieczeniem odpowiednio wykwalifikowanej obsługi operatorskiej. Czas pracy komputera winien być zagwarantowany w terminie od 15 do 7 dni przed terminem odbioru produktu.

Odbiór techniczny produktu

Podstawę odbioru technicznego produktu systemowego stanowią wyniki maszynowego przebiegu testującego, wykonanego na takim zestawie komputerowym, na którym produkt będzie eksploatowany przez nabywcę lub użytkownika. Wyniki te muszą być uzyskane przy zastosowaniu opracowanego przez producenta standardowego modelu danych próbnych.

Odbiór techniczny produktu systemowego następuje w formie protokólarnej przy udziale przedstawicieli nabywcy lub użytkownika oraz OBRI.

USŁUGI SERWISOWE

OBRI świadczy na rzecz nabywcy lub użytkownika swoich produktów programowych pełen zakres usług serwisowych, a mianowicie:

— usuwa wszelkie dostrzeżone w trakcie eksploatacji usterki, zarówno w zakresie programowym produktu, jak i jego dokumentacji technicznej

— dostarcza bezpłatnie użytkownikowi wszelkie uzupełnienia wynikające z doświadczeń z eksploatacji produktu u innych jego użytkowników

— informuje użytkownika o doświadczeniach eksploatacyjnych uzyskanych przy eksploatacji produktu programowego u wszystkich jego użytkowników

— świadczy odpłatnie usługi w zakresie doradztwa organizacyjnego w sferze spraw związanych z funkcjonowaniem produktu

— na życzenie użytkownika opracowuje odpłatnie oprogramowanie uzupełniające w zakresie zastosowań użytkowych oraz wykonuje inne prace związane z wdrażaniem produktu, wykraczające poza zakres usług standardowych

— informuje na bieżąco użytkownika o wszelkich modyfikacjach i kierunkach rozwoju zakupionego produktu programowego.

Skorzystaj z propozycji Ośrodka Badawczo-Rozwojowego Informatyki!

OBRI oferuje następujące produkty programowe:

- dla komputerów serii ODRA 1300 i ICL 1900 — PLANTYP, SEIK, OPTY, KONFORM, JAZ, SOWIK, KTM
- dla komputerów JS EMC (RIAD) i IBM 360/370 — STEP, LP-SERWIS, RODAN, SYKON

Bliższe informacje i szczegółowe oferty można uzyskać w Zakładzie Obrotu Oprogramowaniem i Doradztwa Informatycznego OBRI (ul. Skrońskiego 7, 02-466 Warszawa; tel. 23-77-81 lub 23-71-69) oraz w najbliższym Zakładzie Elektronicznej Techniki Obliczeniowej ZETO

Współpraca z jednostkami resortu finansów

Okres szczególnie intensywnej współpracy pomiędzy jednostkami resortu finansów a przedsiębiorstwami Zjednoczenia Informatyki nastąpił po podpisaniu (w dniu 29 października 1976 r.) porozumienia o współpracy Ministerstwa Finansów i Ministerstwa Nauki, Szkolnictwa Wyższego i Techniki. Chociaż lata poprzednie charakteryzowały się także znacznym udziałem usług ZETO świadczonych jednostkom resortu finansów, to wspomniane porozumienie stworzyło lepsze warunki techniczne i ekonomiczne jednostkom resortu finansów do wdrażania i rozwijania systemów informatycznych w sieci terenowych ZETO w odniesieniu do tych prac, które nie będą wymagały przetwarzania we własnych ośrodkach obliczeniowych. W myśl porozumienia Ministerstwo Finansów będzie także partycypowało w nakładach inwestycyjnych Zjednoczenia Informatyki na budowę i rozbudowę sieci ZETO.

Przedsiębiorstwa ZETO opracowują i eksploatują systemy informatyczne na potrzeby następujących jednostek organizacyjnych:

- Narodowego Banku Polskiego wraz z Powszechną Kasą Oszczędności
- Banku Gospodarki Żywnościowej
- Banku Polska Kasa Opieki SA
- Państwowego Zakładu Ubezpieczeń.

JEDNOSTKI WIODĄCE

W konsekwencji zawarcia porozumienia Zjednoczenie Informatyki powierzyło funkcję przedsiębiorstwa wiodącego Zakładowi Elektronicznej Techniki Obliczeniowej w Poznaniu, a ponadto postawiło do dyspozycji kontrahenta potencjał produkcyjny i badawczo - rozwojowy pozostałych przedsiębiorstw, obejmujący pełny zakres usług (projektowanie, wdrażanie, eksploatacja oraz konserwacja systemów informatycznych). W resorcie finansów rolę jednostki wiodącej powierzono branżowemu Ośrodkowi Badawczo-Rozwojowemu Informatyki Finansowej (OBRIF).

ZETO Poznań — w ścisłej współpracy z OBRIF oraz Departamentem Informatyki NBP — dokonuje uzgodnień projektów, wdrożeń i upowszechnień nowych lub już eksploatowanych systemów informatycznych z jednostkami informatycznymi resortu finansów (np. z Centrum Elektronicznym NBP, Oddziałem Opracowań Maszynowych PZU itp.). Wspólnie prowadzona jest również pełna ewidencja systemów opracowywanych i eksploatowanych przez jednostki resortu finansów — zarówno w sieci ZETO, jak i poza nią.

ZETO Poznań opiniuje także bieżące programy rozwoju usług na rzecz jednostek resortu finansów oraz szczegółowe porozumienia o współpracy, sporządzane przez przedsiębiorstwa Zjednoczenia Informatyki.

Koordynacja taka przyczynia się niewątpliwie do wzrostu jakości świadczonych usług i istotnego przyspieszenia informatyzacji jednostek resortu finansów.

SYSTEMY INFORMATYCZNE RESORTU FINANSÓW

W resorcie finansów istnieje ponad 20 wdrożonych i eksploatowanych systemów informatycznych. Systemy te w większości związane są z bezpośrednią obsługą klienta instytucji bankowych i ubezpieczeniowych, a w związku z tym wymagają bardzo wnikliwej kontroli danych wejściowych, rygorystycznego przestrzegania z góry określonego terminu wykonania obliczeń, jak również pełnej prawidłowości uzyskanych wyników, ponieważ są to czynności dotyczące bieżących operacji pieniężnych.

Spośród realizowanych w ramach porozumienia systemów informatycznych należy wymienić następujące (w kolejności obsługiwanych jednostek resortu finansów):

● Narodowy Bank Polski

System Operacji Bankowych — SOB

System należy do największych i najbardziej złożonych systemów realizowanych obecnie w sieci ZETO. Obejmuje całość prac dotyczących ewidencji rachunkowej i sprawozdawczości w zakresie operacji bankowych. Podstawowe dane zawarte są w dokumentach kasowych i rozliczeniowych dotyczących dyspozycji gotówkowych, bezgotówkowych, walutowych i dewizowych, przekazywanych przez klientów lub sporządzanych przez aparat bankowy. Dane te pozwalają emitować przez komputer wszystkie dokumenty wtórne oraz sporządzać potrzebne sprawozdania dzienne i okresowe.

Przetwarzanie prowadzone jest w cyklach: dziennym, miesięcznym, kwartalnym i rocznym. Szczególnie trudne w realizacji jest przetwarzanie w cyklu dziennym, zapewniające placówkom NBP i ich klientom uzyskanie informacji dotyczących m. in. bieżących obrotów i wyciągów z kont oraz zbiorczych zestawień obrotów, sald i wolnych środków, jak również liczb procentowych, stanowiących podstawę do ustalania wielkości odsetek.

SOB opracowany został pierwotnie na komputer bankowy typu NCR, a następnie zaadaptowany przez ZETO Gdynia na komputer ODRA serii 1300. W sieci ZETO w większości przypadków system przetwarzany jest obecnie na komputerach tej serii. W ostatnim czasie ZETO Katowice przystąpiło do przeniesienia SOB na komputery Jednolitego Systemu.

Na dzień 1 stycznia br. systemem SOB objętych było 106 oddziałów NBP (25,1%), w których w ciągu roku 1977 wykonano 21 715 tys. operacji (44,7% wszystkich operacji NBP).

Krajowy System Ewidencji Rachunków Oszczędnościowych — KSERO

KSERO jest podstawowym systemem eksploatowanym w oddziałach specjalistycznych PKO. Obejmuje on analityczne księgowanie na rachunkach klientów, obliczanie odsetek i drukowanie wyciągów oraz przygotowywanie danych do księgowości syntetycznej i sprawozdawczości. Podstawowe dane zawarte są w dowodach wpłat i wypłat oraz notach memoriałowych. Przetwarzanie prowadzone jest w cyklach: dziennym, dekadowym, miesięcznym i rocznym. Np. w cyklu dziennym uzyskuje się następujące informacje wynikowe:

- zestawienie obrotów i sald
- wykaz zezwoleń na wypłaty z rachunków oszczędnościowych
- wyciągi z kont
- zestawienie różnic na rachunkach oszczędnościowych
- zestawienie sald debetowych rachunków oszczędnościowych.

Podobnie jak SOB system KSERO opracowany został początkowo na komputer bankowy typu NCR, a następnie przeniesiony przez ZETO Gdynia na komputery ODRA serii 1300 oraz przez ZETO Kraków na komputery MIŃSK 32.

O stopniu rozpowszechnienia systemu KSERO, a także systemu ROPS (Rachunki Oszczędnościowe Premiowane Samochodami) świadczy fakt, że systemy te na dzień 1 stycznia br. objęły aż 211 oddziałów PKO (81,1%) oraz 32 115 tys. rachunków oszczędnościowych 78,1% ogółu rachunków oszczędnościowych PKO).

● Bank Gospodarki Żywnościowej

System Zautomatyzowanej Obsługi Długoterminowych Kredytów Inwestycyjnych Ludności Wiejskiej — ZODKILW

ZODKILW obejmuje księgowanie operacji związanych z windykacją udzielonych kredytów oraz przygotowywanie danych do sprawozdawczości kwartalnej i rocznej. Dane wejściowe zawarte są na kartach księgowych, wezwaniach płatniczych i dokumentach aktualizujących. Przetwarzanie odbywa się w okresach miesięcznych z wyraźnie występującymi szczytami w kwietniu i wrześniu. System, opracowany pierwotnie na komputer NCR, został następnie zaadaptowany przez ZETO Łódź na komputery ODRA serii 1300.

● Bank Polska Kasa Opieki — SA

System Ewidencji Rachunków Walutowych Oprocentowanych REWAL

Główne funkcje systemu:

- księgowanie zaszłości na poszczególnych rachunkach walutowych
- dokonywanie zmian pozaksięgowych w kartotece księgowej
- formalna i merytoryczna kontrola dokonywanych operacji
- naliczanie i kapitalizacja odsetek oraz bieżące korygowanie liczb procentowych — system automatycznie

bada naruszenie warunków umowy i w takich przypadkach nalicza karne odsetki
— sporządzanie zestawień wynikowych, dziennych i okresowych
— sporządzanie syntetycznych zestawień obrotów kont przeciwstawnych korespondujących z kosztami rachunków walutowych.

System eksploatowany jest na komputerach ODRA 1300. Wykonawcą systemu jest ZETO Kraków, w którym odbywa się pilotowe wdrożenie pełnego systemu. Obsługą objęci są posiadacze oprocentowanych rachunków walutowych Banku PKO SA.

W roku bieżącym przewiduje się rozpowszechnienie systemu w województwach: wrocławskim i łódzkim, w 1979 r. — poznańskie i gdańskie (biorąc pod uwagę poprzedni podział terytorialny kraju).

● **Państwowy Zakład Ubezpieczeń**
System obliczania odszkodowań za szkody masowe w uprawach gospodarstw indywidualnych

System obejmuje ewidencjonowanie szkód masowych w uprawach rolnych gospodarstw indywidualnych w zakresie ubezpieczeń obowiązkowych i dobrowolnych oraz kontraktacji. Przetwarzanie realizuje następujące funkcje:
— obliczanie odszkodowań na podstawie wykazu szkód sporządzanych przez inspektoraty PZU w wyniku zaistniałych klęsk żywiołowych
— sporządzanie zbiorczych wykazów odszkodowań oraz indywidualnych zleceń wypłaty
— sporządzanie rejestrów odszkodowań wypłaconych i odmówionych
— sporządzanie sprawozdań rocznych.

Dane przenoszone są z wykazów szkód sporządzanych przez inspektoraty PZU. Częstotliwość przetwarzania jest zdeterminowana liczbą i częstotliwością zaistniałych szkód.

System opracowano w ZETO Poznań na komputer ODRA serii 1300. Przewiduje się, że w przyszłości będzie on jednym z modułów Kompleksowego Systemu Automatyzacji Ubezpieczeń Rolnych i Świadczeń Płatniczych.

Kompleksowy System Automatyzacji Ubezpieczeń Rolnych i Świadczeń Płatniczych

W oparciu o zintegrowaną bazę danych będą realizowane podstawowe funkcje systemu, określone w następujących podsystemach:
— podatki i opłaty
— ubezpieczenia rolne
— sprzedaż produktów rolnych.

W ramach podsystemu Podatki i Opłaty dokonywany będzie wymiar i rozliczanie zobowiązań pieniężnych od rolników i ludności zamieszkałej na terenie gminy.

Podsystem Ubezpieczenia Rolne będzie obejmować ewidencję ubezpieczeniowo-szkodową, obliczanie odszkodowań w ubezpieczeniu upraw oraz sprawozdawczość i analizy przebiegu szkodowości w budynkach, mieniu ruchomym, uprawach i zwierzętach.

Podsystem Sprzedaż Produktów Rolnych przeznaczony będzie do ewidencji sprzedanych przez rolników produktów rolnych, dla celów ustalania świadczeń emerytalnych i rentowych.
System opracowuje ZETO Poznań. Planowane wdrożenie systemu nastąpi w roku 1980 w województwie poznańskim, a następnie planuje się dalsze jego sukcesywne upowszechnianie.

System Jednostkowych Rent Odroczonych — RENO

System RENO realizuje następujące funkcje:
— ewidencjonowanie ubezpieczeń rent odroczonych
— kontrolę prawidłowości naliczania renty, zwrotów składek, skapitalizowanej wartości renty oraz stanu renty na koniec roku
— obliczanie rezerw techniczno-ubezpieczeniowych rent odroczonych
— sporządzanie zestawień statystycznych dla inspektorów, oddziałów wojewódzkich i centrali PZU.

System opracowany został przez ZETO Wrocław na komputery ODRA serii 1300.

ZAPOTRZEBOWANIE NA USŁUGI

Niezależnie od eksploatacji opisanych systemów uzgodnione plany na lata 1978—1980 przewidują, że ośrodki ZETO będą prowadziły prace nad dalszą adaptacją i upowszechnieniem niektórych systemów już użytkowanych, a także będą opracowywały, wdrażały i rozpowszechniały nowe systemy informatyczne na użytek jednostek resortu finansów.

Program tych prac dotyczy 5 jednostek resortu finansów, przy czym ich wykonawcami jest 6 ośrodków ZETO. Spośród 7 tematów — 4 przewidziane są do wdrożenia w 1978 r., jeden w 1980 r., natomiast pozostałe — w 1979 r.

Zapotrzebowanie na usługi informatyczne realizowane w sieci ZETO dla instytucji resortu finansów przedstawia poniższa tabela, przy czym w pozycji „NBP" zawarte są dane zarówno z NBP, jak i PKO. Wielkości ujęte w tabeli wyrażają zapotrzebowanie na usługi obliczeniowe w godzinach pracy komputerów oraz procentowo porównywane z wielkościami wyjściowymi przyjętymi za 100. Zestawienie tabelaryczne obejmuje realizację z lat 1976—77 oraz planowane zapotrzebowanie na usługi w latach 1978—80.

Z przedstawionych danych wynika stały znaczny wzrost łącznego zapotrzebowania usług obliczeniowych przebiegających jednak u poszczególnych kontrahentów bardzo nierównomiernie.

Zapotrzebowanie na usługi informatyczne w sieci ZETO przez jednostki resortu finansów

Lp.	Zleceniodawca (użytkownik systemu)	Lata				
		Wykonanie		Plan		
		1976	1977	1978	1979	1980
	Usługi obliczeniowe — godziny komputerów / wskaźnik przyrostu					
1	NBP	72 763 / 100,0	90 258 / 124,0	93 177 / 128,0	96 768 / 133,0	111 490 / 153,2
2	BGŻ	— / —	2 880 / 100,0	2670 / 92,7	3040 / 105,5	6040 / 209,7
3	Bank PKO SA	— / —	1090 / 100,0	480 / 44,0	480 / 44,0	480 / 44,0
4	PZU	5 / —	372 / 100,0	580 / 155,0	9800 / 2634,4	11 020 / 2962,3
	Razem	72 768 / 100,0	94 600 / 130,0	96 907 / 133,2	110 088 / 151,3	129 030 / 177,3
	Usługi projektowania, programowania i doradztwa — godziny specjalistów / wskaźnik przyrostu					
1	NBP	29 227 / 100,0	29 882 / 102,2	6480 / 22,2	6880 / 23,5	7500 / 25,7
2	BGŻ	6126 / 100,0	6314 / 103,0	500 / 8,2	2500 / 40,8	1900 / 31,0
3	Bank PKO SA	9548 / 100,0	10 097 / 105,7	500 / 5,2	500 / 5,2	500 / 5,2
4	PZU	13 708 / 100,0	16 517 / 120,5	31 100 / 226,8	6000 / 43,8	100 / 0,7
	Razem	58 609 / 100,0	62 810 / 107,2	38 580 / 65,8	15 880 / 27,1	10 000 / 17,1

MOŻLIWOŚCI DALSZEGO UPOWSZECHNIENIA SYSTEMÓW

Obecny stan eksploatacji systemów informatycznych na potrzeby resortu finansów przez poszczególne ośrodki ZETO, jak również osiągnięty w tym zakresie postęp zilustrowano na rysunku.

JEDNOSTKI ZETO	SYSTEMY							
	SOB (NBP)	KSERO (NBP / PKO)	R-KI OSZCZĘDNOŚCIOWE (NBP/PKO)	ZODKILW (BGŻ)	SZKODY W UPRAWACH (PZU)	REWAL (PKO SA)	ROPS (NPB/PKO)	RENO (PZU)
Białystok	●	●		●				
Bydgoszcz *)								
Gdynia	●	●						
Olsztyn	●	●						
Jelenia Góra	●	●						●
Kielce	●	●		●				●
Katowice **)	■	■	■					■
Bielsko Biała			■					
Częstochowa			■					
Będzin								
Dąbrowa Górnicza								
Koszalin	●	●	●					
Kraków **)	●	●	●	●		●	●	
Rzeszów	●	●		●				
Tarnów								
Krosno								
Lublin	●	●		●				
Łódź	●	●		●				
Poznań	●	●		●	●			
Piła								
Szczecin	●	●						
Warszawa *)								
Wrocław	●							●
Opole	●	●						
Świdnica		●		●			●	
Zielona Góra	●	●						

Eksploatacja na komputerach:
●- ODRA serii 1300
■- JS EMC

Eksploatacja systemów informatycznych dla jednostek resortu finansów w ośrodkach ZETO

*) W Bydgoszczy i Warszawie istnieją ośrodki obliczeniowe NBP

**) W Katowicach i Krakowie przetwarzanie odbywa się również na komputerach MIŃSK-32

Dane obejmują okres do końca 1977 r. i jak wynika z rysunku najbardziej rozpowszechniona jest eksploatacja systemów SOB i KSERO, natomiast pozostałe systemy są eksploatowane w mniejszej liczbie ośrodków. Wnioski wynikające z tego rysunku dla ZETO Poznań jako przedsiębiorstwa wiodącego w realizacji porozumienia są podstawą do zintensyfikowania wdrożeń i upowszechnienia istniejących rozwiązań w dalszych jednostkach resortu finansów. Możliwości i potrzeby dalszego wdrażania systemów w następnych ośrodkach ZETO są bezsporne. Należy zwrócić szczególną uwagę na możliwości upowszechnienia takich eksploatowanych już systemów, jak:
— System Zautomatyzowanej Obsługi Kredytów Inwestycyjnych Ludności Wiejskiej — ZODKILW dla BGŻ

— system obliczania odszkodowań za szkody masowe w uprawach gospodarstw indywidualnych dla PZU

— system ewidencji rachunków walutowych oprocentowanych REWAL dla Banku PKO SA

— Rachunki Oszczędnościowe Premiowane Samochodami ROPS dla PKO

— system jednostkowych rent odroczonych RENO dla PZU.

Poza wymienionymi systemami istnieją również znaczne możliwości dalszego upowszechniania systemów SOB i KSERO w oddziałach NBP nie objętych dotąd automatyzacją przetwarzania.

PERSPEKTYWY I KORZYŚCI DALSZEJ WSPÓŁPRACY

Spośród wszystkich resortów korzystających z usług sieci ZETO resort finansów był największym kontrahentem, wykorzystującym w latach 1976 i 1977 ok. 18% ogółu usług świadczonych przez przedsiębiorstwa Zjednoczenia Informatyki. Pozostała część potencjału usług rozkładała się na około 30 innych resortów.

W planach na najbliższe lata przewiduje się dalszą ścisłą współpracę przedsiębiorstw ZETO z resortem finansów w adaptacji istniejących i tworzeniu nowych systemów oraz ich wdrażaniu, upowszechnianiu i bieżącej eksploatacji. W związku ze stopniowym przechodzeniem na eksploatację komputerów Jednolitego Systemu EMC należy w większym niż dotychczas stopniu dążyć do utworzenia lub adaptacji systemów ukierunkowanych na te komputery, a także do zautomatyzowania coraz większej liczby jednostek resortu finansów. Automatyzacja operacji finansowych prowadzi bowiem do znacznego skrócenia cykli rozliczeniowo-ewidencyjnych, zmniejszenia zatrudnienia w komórkach rozliczeniowo-ewidencyjnych oraz bardziej terminowej obsługi setek tysięcy, a nawet milionów (PKO) klientów jednostek resortu finansów.

Mgr Jerzy BEDNARZ
Mgr. inż. Zenon MNISZAK
ZETO Poznań

Białostockie Dni Informatyki

W ramach XI Białostockich Dni Techniki (16—26 października br.) odbędą się Dni Informatyki, zorganizowane przez Międzywojewódzki Komitet Naukowo-Techniczny NOT ds. Informatyki przy współudziale SKwP, SITR i ZETO Białystok.

Podstawową imprezą Dni Informatyki będzie **II Regionalna Konferencja Informatyków** (19 października), obejmująca swym zasięgiem województwa: białostockie, łomżyńskie i suwalskie. Temat Konferencji — „Informatyka w kompleksie gospodarki żywnościowej" — powinien zainteresować również informatyków spoza wymienionego obszaru.

Poza tym przewidziane są seminaria (tematy: „Typowe moduły informatyczne w przedsiębiorstwie", „Systemy zarządzania bazą danych i struktura baz danych"), odczyty (tematy: „Transmisja danych — oprogramowanie i sprzęt", „Współczesne metody tworzenia maszynowych nośników informacji — koncentratory danych", „Transmisja i konwersja danych przy współpracy EMC ODRA i RIAD"), wystawa publikacji z dziedziny informatyki.

Organizatorzy II Regionalnej Konferencji Informatyków serdecznie zapraszają do udziału w imprezie. Zgłoszenia należy kierować do 30 sierpnia br. pod adresem: Oddział Wojewódzki NOT w Białymstoku, ul. M. Curie-Skłodowskiej 2. Można również zgłosić komunikaty dotyczące zastosowań informatyki w gospodarce żywnościowej. **(A.O.)**

Zarząd Oddziału Warszawskiego Stowarzyszenia Socjalistycznego Marketingu i Zjednoczenie Informatyki organizują sympozjum pod hasłem

„Informatyka na usługach marketingu".

Sympozjum odbędzie się w październiku br. w Warszawie.

Sympozjum jest przeznaczone głównie dla kadry kierowniczej organizacji przemysłowych i handlowych wszystkich branż działających w obrocie krajowym i zagranicznym.

Celem sympozjum jest omówienie roli informacji i informatyki w działalności marketingowej, prezentacja skomputeryzowanych systemów informacji marketingowej dla naczelnego kierownictwa (na przykładzie resortu handlu zagranicznego) oraz wymiana doświadczeń i opinii na temat stosowania i wdrażania informatyki do działań marketingowych w kraju i za granicą.

Wszelkich informacji o sympozjum udziela Stowarzyszenie Socjalistycznego Marketingu, ul. Nowy Świat 49, 00-042 Warszawa, telefon 27-62-39.

Jeszcze o nartach

Długi cykl wydawniczy naszego pisma spowodował, że dopiero w pełni sezonu żeglarskiego możemy zdać relację z imprezy marcowej: narciarskich mistrzostw informatyków. Odbywają się one co roku, tym razem już po raz szósty. Stale rośnie liczba uczestników: w pierwszych startowało niecałe trzydzieści osób, w ostatnich ponad sto.

Pierwszą konkurencję — slalom gigant — rozegrano na stoku Hali Szrenickiej. W klasyfikacji zespołowej wygrało ZETO Jelenia Góra, drugie miejsce zajęło ELWRO, a trzecie ZETO Wrocław. Na kolejnych pozycjach sklasyfikowano: ZIPO Gdańsk, ZETO Katowice, ZETO Jelenia Góra (II reprezentacja), IMM Warszawa, MERA-ELZAB Zabrze, OBRI Warszawa i ZETO Poznań.

Niespodzianką było przełamanie hegemonii Dolnego Śląska w konkurencjach kobiecych. Zarówno w slalomie gigancie, jak i w rozgrywanym następnego dnia slalomie specjalnym najlepsza była Magdalena Czaińska z IMM Warszawa.

Zawody odbyły się w dość trudnych warunkach atmosferycznych, ale zła pogoda nie odstraszyła informatyków. Nie złamały ich także warunki mieszkaniowe. Należy to zawdzięczać tylko energii przewodniczącej komitetu organizacyjnego, mgr Elżbiety Mrozowicz z ZETO Wrocław, której udało się jednak pomieścić taką masę ludzi w ośrodku Stoczni im. Komuny Paryskiej w Szklarskiej Porębie.

Nie wytrzymał natomiast sprzęt informatyczny. Przywiezione przez ekipę ELWRO urządzenie do błyskawicznego podawania wyników przejechało się wyciągiem narciarskim, zmarzło i nie chciało funkcjonować. Na nic więc nie przydała się znakomita drukarka ze standardowego wyposażenia kalkulatorów. Zdaniem projektantów tego urządzenia przyczyną niepowodzenia była zła jakość układów scalonych z CEMI. Epizodu tego nie należy oczywiście traktować jako symbolicznego obrazu polskiej informatyki.

Narciarskie mistrzostwa informatyków — jedyna w naszym środowisku sportowa impreza zimowa — znakomicie uzupełniają całoroczny program rekreacyjny i cieszą się nadspodziewanie dużym zainteresowaniem. Może więc przydałaby się jeszcze jakaś impreza zimowa (niekoniecznie narciarska)... **(mh)**

Przed pierwszym setem

Stosunki międzyludzkie w światku informatycznym nie muszą być najgorsze, skoro dyscypliny sportowe, w szranki których pozywają się jak dotychczas jego przedstawiciele, są łagodniejsze niż judo, boks czy karate.

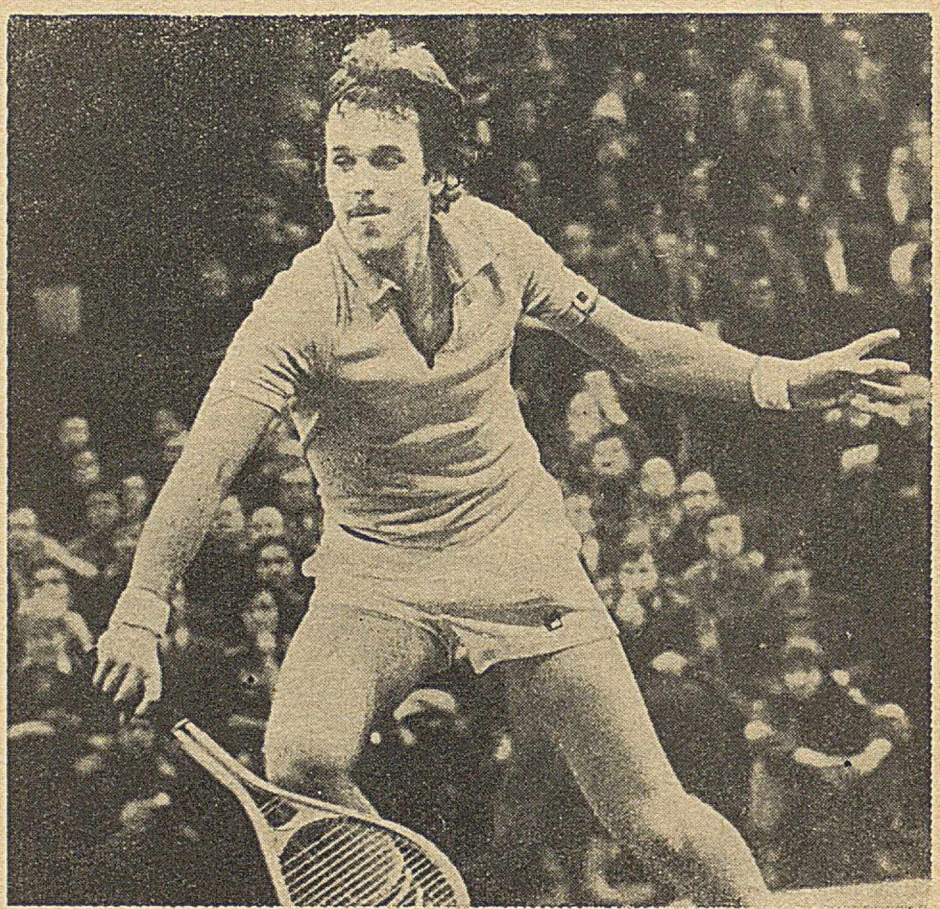
Po tradycyjnych już mistrzostwach narciarzy, po regatach i siatkówce, zaprezentują swoje umiejętności tenisiści. Zamiar organizacji mistrzostw tenisowych informatyków — poprzedzony głębokim sondażem, który dowiódł, że Fibak jest tak samo znany w środowisku jak COBOL — konkretyzuje się na przełomie sierpnia i września na kortach Jeleniej Góry, dzięki tamtejszym zapaleńcom oraz życzliwemu stanowisku dyrekcji ZETO i Zjednoczenia Informatyki.

Wszelako okazało się, że turniej zakrojony przecież na dość dużą skalę (czas trwania — 4 dni, 48 uczestników płci obojga i 6 konkurencji) nie zadowoli jednak wszystkich chętnych — gdyby miał zadowolić, musiałby trwać parę tygodni.

W przyszłości trzeba więc będzie rozgrywać mistrzostwa strefowe — eliminacyjne, po których tylko najlepsi zmieszczą się w ramach organizacyjnych finału. Zwłaszcza jeżeli mają to być rzeczywiście mistrzostwa informatyków, a nie tylko pracowników podległych Zjednoczeniu Informatyki (choć na chwałę organizatorów należy zauważyć, że informatyków spoza sieci ZETO starają się nie dyskryminować).

Bez względu na sceptyczne komentarze krążące z racji rozmaitych „mistrzostw informatyków" inicjatywa ZETO Jelenia Góra jest na pewno cenna. Niejednokrotnie akcentowaliśmy korzyści płynące ze swobodnych towarzyskich i sportowych kontaktów, a przede wszystkim ich funkcję integracyjną.

A może by udało się jakoś wyróżnić tenisowe mistrzostwa informatyków od mistrzostw budowlanych czy kolejarzy? Ot, na przykład przy okazji jakiś mały konkurs na algorytm systemu organizacji rozkładu gier dla poszczególnych konkurencji (byłyby i dodatkowe walory do punktacji)? Albo projekt komputerowego systemu klasyfikacji dla krajowych wyczynowców (na wzór komputerowej listy WCT)? Obok elementu ćwiczenia i zabawy może być z tego pożytek. Może nawet ktoś (PZT?) to kupi? **(KB)**



Wszystkim kibicom życzymy ciekawych wrażeń, zawodnikom — zręczności Fibaka

Komputer w szkole

Jaka powinna być informatyka w szkole? Jaka na uczelniach technicznych i humanistycznych? Słowem, jaki powinien być kompleksowy system nauczania informatyki w naszym kraju na obecnym etapie rozwoju sprzętu komputerowego, w perspektywie powszechnego zastosowania automatycznego przetwarzania informacji na użytek społeczeństwa? Są to chyba, w chwili obecnej, najważniejsze kwestie, które będą decydować o rozwoju zastosowań w niedalekiej już przyszłości. Pytania te pozostają dotychczas bez jednoznacznej odpowiedzi; specjaliści od nauki i dydaktyki toczą spory, czy szkołom „dać" komputery czy nie, a tymczasem życie stwarza coraz to nowe sytuacje.

Przykładem może być ośrodek obliczeniowy, który został otwarty przy Zespole Szkół Elektronicznych w Bydgoszczy. Ma już nie byle jaką maszynę, bo ODRĘ 1304, którą wysiłkiem wielu osób zdołano doprowadzić do stanu używalności i uruchomić w II kwartale tego roku. A więc wcześniej niż resort oświaty mógł zapewnić tym szkołom taką inwestycję. Komputer ten, mający na liczniku już 4 tysiące godzin pracy, został przywieziony do Bydgoszczy w bardzo złym stanie. Uruchomienie go wysiłkiem kadry pedagogicznej zatrudnionej w ZSE nie byłoby możliwe. Dlatego mgr Jerzy Piłat, zastępca dyrektora, który doprowadził do praktycznego sfinalizowania prac związanych z instalacją maszyny, z najwyższym uznaniem wyraża się o specjalistach z ZETO w Bydgoszczy i Ośrodka Informatyki Pomorskiego Okręgu Wojskowego. Oni to właśnie doprowadzili do zakończenia tych prac w ciągu trzech miesięcy.

Patronat nad uruchomieniem komputera objęło ZSMP przy ZETO, a szczególnie wyróżnili się inż. Jerzy Kociszewski, inż. Andrzej Paliwoda i inż. Zbigniew Sinda. Z Ośrodka Informatyki POW największy wkład pracy jest udziałem Adama Lisa.

A braków w maszynie było wiele. Brakowało między innymi kabli, brakowało także całych podzespołów. Wiele z tych brakujących elementów i części otrzymano z innych ośrodków. W krótkim czasie adaptowano pomieszczenia umożliwiające zainstalowanie komputera.

W Zespole Szkół Eelektronicznych istnieje również studium pomaturalne, gdzie jednym z kierunków jest programowanie maszyn cyfrowych. Od roku 1975 do zajęć wykorzystywano mały komputer typu ODRA 1103, natomiast w pracach dyplomowych trzeba było korzystać z komputerów zainstalowanych w innych ośrodkach. Po uruchomieniu „czwórki", uczniowie będą mogli więc „pisać" prace te w szkole. Przewiduje się także utworzenie nowego kierunku. Dyrekcja szkół zapewnia, że z komputera będą mogły korzystać także inne szkoły województwa bydgoskiego. Ośrodek obliczeniowy, który powstał przy Zespole Szkół Elektronicznych, powinien więc stać się ośrodkiem międzyszkolnym. W ścisłej współpracy z ZETO Bydgoszcz może on także przynosić resortowi pewien dochód z eksploatacji w czasie wolnym systemów informatycznych dla miejscowych przedsiębiorstw.

Uroczyste otwarcie Ośrodka odbyło się wprawdzie w największej sali szkolnej, jednakże z braku miejsc nie wszyscy uczniowie mogli w nim uczestniczyć bezpośrednio.

Na uroczystość otwarcia przybyli przedstawiciele ośrodków współpracujących przy uruchamianiu maszyny oraz zaproszeni goście z regionu bydgoskiego. Inauguracyjny referat został wygłoszony przez dyrektora Uczelnianego Ośrodka Obliczeniowego przy Uniwersytecie Mikołaja Kopernika w Toruniu, dr. Bronisława Żurawskiego. Temat: „Rola informatyki we współczesnym społeczeństwie i zadania szkół w tej dziedzinie" zapowiadał przedstawienie wielu interesujących problemów. Referent skoncentrował się jednakże na wygłoszeniu apelu do młodzieży o przychylne spojrzenie na informatykę, o zainteresowanie się możliwościami wykorzystania jej w codziennym życiu. Forma i treść referatu nie odbiegały od wykładów, jakie słyszy zwykle uczeń liceum, co wprawdzie nie było w sprzeczności z miejscem wygłaszania, jednakże w jaskrawej sprzeczności z zapowiadaną treścią.

W drugiej części programu odbyła się prezentacja ośrodka i uroczyste uruchomienie komputera. Jako pierwsi obsługiwali maszynę harcerze z drużyny szkolnej. Następnie goście obejrzeli wystawę prac dyplomowych, wykonanych przez uczniów technikum, na której można było zobaczyć między innymi model dydaktyczny „kalkulatora elektronicznego K-741 M", wykonany przez Sławomira Kulkę i Wojciecha Karczewskiego, oraz „układ do współpracy dalekopisu z magnetofonem", który eliminuje taśmę pepierową.

Dyrekcja szkół liczy na to, że w najbliższym czasie będzie mogła uzupełnić zestaw posiadanego sprzętu jednostkami pamięci taśmowej PT3 oraz alfanumerycznymi dziurkarkami i sprawdzarkami kart. Obecnie szkoła przystępuje do realizacji rozszerzonych ambitnych zadań, wychodząc naprzeciw potrzebom regionu i postulatom społeczeństwa. Niedługo więc będziemy mogli znowu zobaczyć, jak praktyka i konkretne działanie wyprzedzają teorię.

Tekst i zdjęcie: Andrzej KLIMEK

Pokaz sprzętu pomocniczego

W dniach od 12 do 14 kwietnia br. wiedeńska firma EQUIPEX, reprezentująca trzech czołowych producentów sprzętu pomocniczego, zorganizowała w salach Hotelu Europejskiego w Warszawie interesujący pokaz nowości. Trzeba stwierdzić, że postęp techniczny przebiega tu równie szybko, jak w dziedzinie sprzętu podstawowego, choć jest on u nas prawie niedostrzegalny ze względu na bardziej niż skromne nasycenie tego rodzaju urządzeniami przytłaczającej większości krajowych ośrodków obliczeniowych. Sytuacja taka utrzymuje się mimo powszechnej świadomości, że sprzęt pomocniczy w znacznej liczbie przypadków decyduje o efektywności wykorzystania tak kosztownego przecież sprzętu podstawowego. Na wspomniane niedoinwestowanie w dziedzinie sprzętu pomocniczego wpływa w znacznej mierze fałszywie pojęta oszczędność w projektowaniu i zatwierdzaniu projektów ośrodków obliczeniowych, a także brak odpowiednich ofert ze strony producentów krajowych.

Obowiązujący postulat stałego podnoszenia efektywności wykorzystania sprzętu komputerowego zobowiązuje nas do uważnego śledzenia postępu technicznego również w dziedzinie rozwoju sprzętu pomocniczego za granicą. Doskonałą okazją do tego był właśnie pokaz firmy EQUIPEX, pozwalający znacznej liczbie zainteresowanych zapoznać się bezpośrednio z działaniem najnowszych rozwiązań w podstawowych grupach sprzętu pomocniczego.

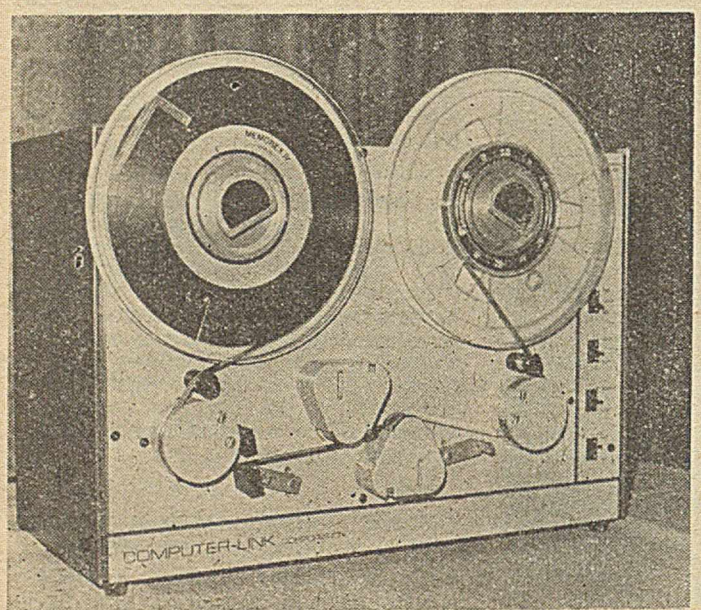
Trzej biorący udział w pokazie producenci sprzętu pomocniczego, to RFN-owskie firmy LAMPERTZ I BÖWE oraz amerykańska firma COMPUTER-LINK Corp. Firmy te specjalizują się w produkcji trzech podstawowych grup sprzętu pomocniczego:

- urządzeń do przechowywania i transportu nośników informacji (LAMPERTZ)
- urządzeń do obróbki wydruków komputerowych (BÖWE)
- urządzeń do regeneracji i kontroli stanu technicznego taśm magnetycznych i pakietów dyskowych (COMPUTER-LINK Corp.).

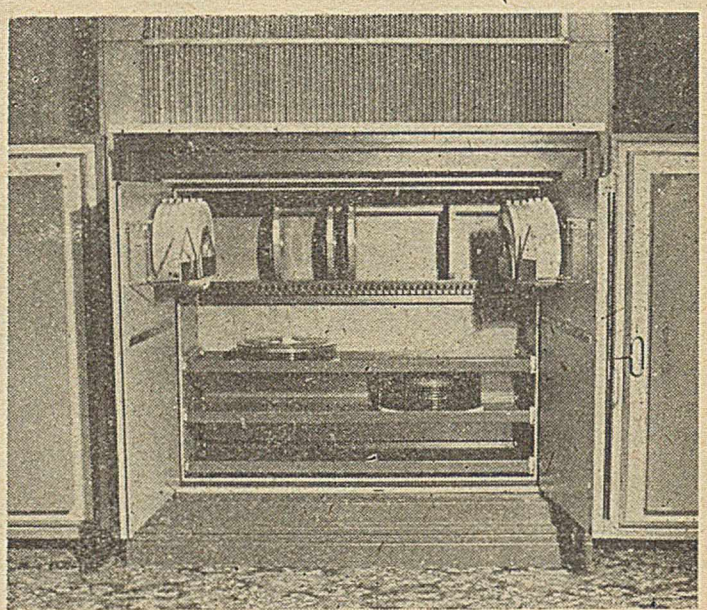
Ze względu na coraz bardziej akcentowane znaczenie problemu zabezpieczenia danych przed zniszczeniem lub uszkodzeniem, głównym akcentem ekspozycji wyrobów firmy LAMPERTZ była ognioodorna szafa do przechowywania magnetycznych nośników danych. Szafy tego typu uzyskały w RFN atest na podstawie przeprowadzonych prób, symulujących rzeczywiste warunki pożaru i zawalenia się stropu (nagrzewanie w temperaturze powyżej 1000°C, zrzucenie z wysokości ok. 10 m), po których przechowywane wewnątrz szafy nośniki magnetyczne zachowały pełną przydatność eksploatacyjną. Oprócz tego szafy te, jak również inne szafy bez zabezpieczeń termicznych, wyposażone są w zamki stosowane w skarbcach bankowych, zabezpieczające przed dostępem do przechowywanych danych osób nieupoważnionych. Zabezpieczenia tego rodzaju są w pełni uzasadnione, jeśli się zważy, że utworzenie, oczyszczenie z błędów i doprowadzenie do aktualności bazy danych w wielkim przedsiębiorstwie stanowi olbrzymi koszt, przekraczający często wartość całego zainstalowanego sprzętu, nie mówiąc już o niewymiernej, bardzo subiektywnej wartości tych danych z punktu widzenia uwarunkowania od nich całej operatywnej działalności tego przedsiębiorstwa. Jeśli chodzi o szafę ogniotrwałą, to dużym zaskoczeniem jest stosunkowo mały ciężar, umożliwiający jej transport bez stosowania specjalnych dźwigów oraz wzmacniania stropów. Swą lekkość szafa ta zawdzięcza zastosowaniu najnowocześniejszych materiałów termoizolacyjnych.

Ekspozycję wyrobów firmy LAMPERTZ uzupełniał szeroki asortyment urządzeń do transportu różnych rodzajów nośników danych, zarówno wewnątrz ośrodka, jak i poza nim (zasobniki do przesyłania pocztą ekspresową). Postęp w konstrukcji wszystkich tych stosunkowo prostych urządzeń wyraża się w dziesiątkach drobnych ulepszeń, ułatwiających i zwiększających wydajność prac pomocniczych w każdym ośrodku obliczeniowym.

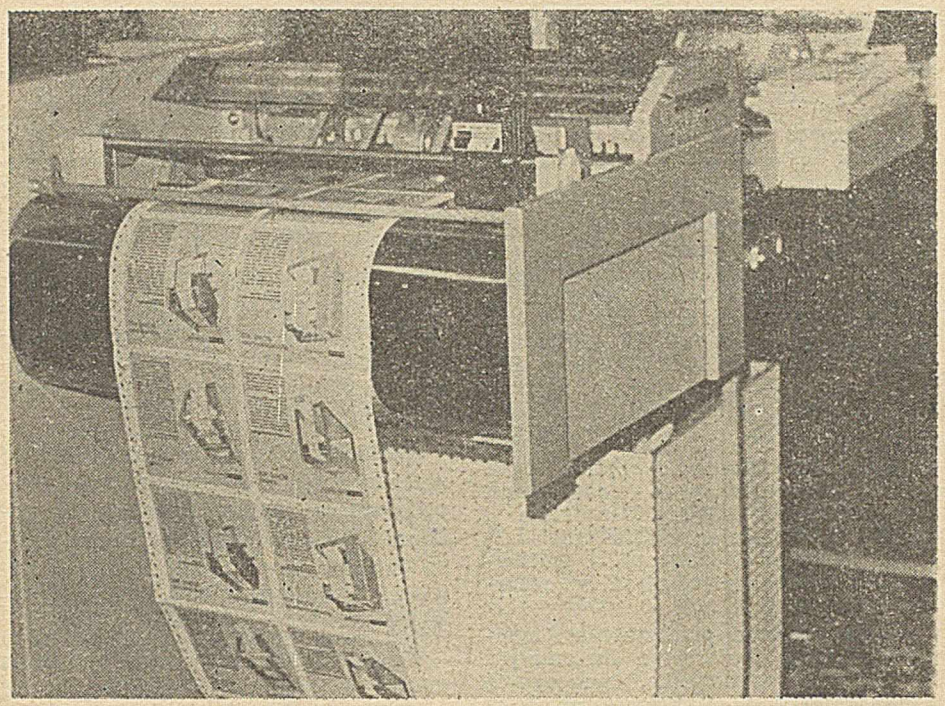
Głównym akcentem ekspozycji wyrobów firmy BÖWE był najnowszy model przecinarki wydruków komputerowych, współpracujący z rozdzielaczem tych wydruków. Przecinarka ta określana jest przez producenta jako najszybsze z oferowanych na rynku światowym urządzeń tego typu. Warto podkreślić, że szybkość działania tego modelu jest przeszło dwukrotnie większa od produkowanego równolegle poprzedniego modelu 303. Wydajność taką udało się osiągnąć dzięki zastosowaniu do sterowania nowoczesnych układów scalonych, zapewniających również olbrzymią precyzję działania przecinarki. Nowy model można rozbudowywać pod względem zakresu wykonywanych funkcji przez przyłączanie programowanego optycznego czytnika znaków kreskowych, znacznie rozszerzającego warunki automatycznego selekcjonowania różnych dokumentów wynikowych. Aktualna oferta firmy BÖWE obejmuje ok. 10 modeli przecinarek, rozrywaczy i rozdzielaczy wydruków komputerowych, charakteryzujących się różnymi wydajnościami i cenami, a więc możliwością dostosowywania do bardzo zróżnicowanych potrzeb użytkowników.



Urządzenie do oczyszczania i przewijania taśmy magnetycznej firmy COMPUTER-LINK



Szafa ogniotrwała firmy LAMPERTZ do przechowywania magnetycznych nośników danych



Przecinarka wydruków komputerowych firmy BÖWE

Trzecią i ostatnią część pokazu stanowiły następujące urządzenia produkcji firmy COMPUTER-LINK:

- urządzenie do kontroli stanu technicznego pakietów dyskowych, produkowane w dwóch wersjach, dostosowanych do standardowych pakietów 6- i 11-dyskowych,
- urządzenie do oczyszczania i przewijania taśmy magnetycznej (ang. *cleaner/rewinder*),
- urządzenie do oceny stanu technicznego i oczyszczania taśmy magnetycznej (ang. *evaluator/cleaner*).

Należy podkreślić, że w odróżnieniu od wielu podobnych imprez organizowanych przez firmy zagraniczne, pokaz firmy EQUIREX wyróżniał się bardzo sprawną demonstracją działania poszczególnych urządzeń przez osoby nie tylko doskonale zorientowane w szczegółach konstrukcyjnych i eksploatacyjnych, a co ważniejsze — udzielające wyjaśnień w języku polskim.

Tekst: Władysław KLEPACZ
Zdjęcia: Andrzej KLIMEK

Prawie debiut, a jakże udany

Zainteresowanie informatyką nie przestaje być specjalnością Towarzystwa Naukowej Organizacji i Kierownictwa. Po ciekawych inicjatywach TNOiK-owskich sekcji informatyki w Szczecinie, Gdańsku i Poznaniu przyszła kolej na sekcję zielonogórską. Formalnie zaś sekcja ta liczy sobie nie więcej niż 2 lata. Tak więc, poza ubiegłoroczną, lokalną imprezą, ostatnia konferencja[1]) — o zasięgu ogólnopolskim — była właściwie debiutem organizacyjnym sekcji informatyki i pracowników TNOiK z Zielonej Góry.

Generalnym założeniem organizatorów konferencji była szeroka popularyzacja optymalnych metod wdrażania systemów informatycznych do sfery organizacji i zarządzania przedsiębiorstw. Jeśli chodzi o podkreślenie roli użytkownika w procesie wdrażania systemów informatycznych, program kołobrzeskiej konferencji nie był rewelacyjnym novum. Co nie znaczy, aby dyskusji i wymiany doświadczeń w tym przedmiocie było za dużo.

Aktywny udział użytkowników systemów informatycznych został uznany za niezbędny zgodnie z naturą rzeczy przynależną organizacji i zarządzaniu systemami gospodarczymi. Dostrzeżono, że obiektowy system informatyczny nie może być zainstalowany w przedsiębiorstwie jako kompletny pakiet procedur w jakimś dowolnie wybranym momencie. Co więcej wdrażanie systemu obiektowego w pełnym zakresie nie jest praktycznie możliwe. Przeszkodą jest tu dynamika procesu organizacji przedsiębiorstwa. Informatyzacja jednej ze sfer zarządzania wywołuje zmiany w pozostałych. Informatyzacja całego przedsiębiorstwa musi się zatem odbywać zgodnie z jakimś harmonogramem ustalającym stopniową komputeryzację poszczególnych sfer organizacji i zarządzania oraz ich wzajemną integrację. Tym większa jest rola użytkownika, że na rynku informatycznym pojawia się coraz więcej gotowych powielarnych systemów. Przy dobrym rozeznaniu potrzeb i rozsądnym sterowaniu wyborem i zakupem gotowych podsystemów przedsiębiorstwo może stosunkowo niskim kosztem zautomatyzować poszczególne obszary działalności; zwłaszcza że może liczyć na pomoc doświadczonych informatyków.

Niejako przy okazji zaprezentowano na konferencji najnowsze tendencje w światowej informatyce oraz kilka powielarnych systemów zrealizowanych w ZETO-Zielona Góra. Przy tej ostatniej prezentacji przedstawiono sprawdzony empirycznie proces organizacji wdrożeń systemów. Adresatem prezentacji byli przede wszystkim potencjalni użytkownicy systemów, którzy stanowili znakomitą większość uczestników konferencji i reprezentowali bez mała wszystkie regiony kraju. Swoją drogą tak liczny udział przedstawicieli kadry kierowniczej zakładów w konferencji dowodzi jej rosnącego zainteresowania informatyką i tendencji do racjonalnego stosowania tego kosztownego narzędzia.

Z zadowoleniem można podkreślić, że kadra kierownicza pozbywa się kompleksów, nie kibicuje już obradującym, aktywnie uczestniczy w dyskusjach informatyków, nawet wymusza na nich takie formułowanie problemów i odpowiedzi, które pozwalają na prawdziwie partnerski dialog. Poza tym podobny dialog ułatwia coraz lepszą znajomość informatyki wśród kadry kierowniczej.

Informatycy też już zrozumieli, że bez aktywnego współuczestnictwa użytkowników trudno jest stworzyć doskonały system informatyczny, a gdyby się taka rzecz nawet udała — system wdrażany w nieprzygotowanym miejscu i w niewłaściwym czasie nie gwarantuje pożądanych efektów.

Wygłoszono następujące referaty:

— „Organizacyjne aspekty projektowania systemów informatycznych przedsiębiorstw przemysłowych" (dr inż. Henryk Pietrowski)

— „Wybrane problemy wdrażania systemów informatycznych zarządzania" (mgr Edward Kram)

— „Wdrażanie powielarnych systemów informatycznych" (mgr Janusz Jackowski i mgr Waldemar Kępski)

— „Organizacja wdrażania branżowych systemów informatycznych" (mgr Włodzimierz Rakocy)

— „Organizacja i metody przygotowania kombinatu do wdrażania systemu informatycznego" (mgr Zbigniew Rogaliński)

— „Wpływ informatyki na system zarządzania przedsiębiorstwem" (mgr Mirosław Bencych).

1) Konferencja pod hasłem „Podstawy organizacyjne wdrażania systemów informatycznych" odbyła się w Kołobrzegu, w dniach 15—16 maja br.

Referaty i żywą dyskusję podsumowano w postaci wniosków. A wnioski te, jak każde inne, mają li-tylko charakter „memento" i stanowią kodeks zaleceń istotnych przy konfrontacji z konkretnym problemem, są także wyrazem swoistego consensus między przedstawicielami różnych koncepcji rozwiązań; zasługują jednak na przedstawienie:

- w celu potanienia procesu informatyzacji przedsiębiorstw należy w maksymalnym stopniu korzystać z systemów powielarnych
- popularyzacja systemów powielarnych jest bardziej efektywna poprzez wdrożenie branżowe niż poprzez wdrożenie w poszczególnych przedsiębiorstwach
- ośrodki opracowujące systemy powielarne powinny koordynować prace wzajemnie w celu integracji systemów stanowiących podsystemy w systemie obiektowym
- należy unowocześnić strukturę systemów powielarnych; powinny one mieć możliwość szybkiego dostępu do zgromadzonych informacji, głównie przez uwzględnienie przetwarzania rozproszonego
- ośrodki ETO powinny specjalizować się w produkcji systemów
- należy zdynamizować udział użytkownika w procesach projektowania systemów, a zwłaszcza w procesie tworzenia podstaw organizacyjnych do wdrażania systemów
- projekty zastosowań informatyki muszą być ściśle zintegrowane z systemem organizacji i zarządzania przedsiębiorstwem
- należy dążyć do wdrażania takich systemów, które zapewnią informatyzację wszystkich dziedzin działalności przedsiębiorstwa
- w celu sprawnej eksploatacji systemów niezbędna jest szybka dostawa na rynek urządzeń do wprowadzania danych na taśmę magnetyczną
- systemy powielarne powinny być konstruowane tak, aby eksploatacja była możliwa zarówno przy korzystaniu z papierowych nośników danych, jak i z magnetycznych
- proces projektowania, szczególnie na poziomie logicznym systemu, powinien w dużym stopniu opierać się na technice komputerowej (symulacja).

Wnioski obejmują więc wiele zagadnień — od metod projektowania i programowania, poprzez udział i rolę użytkownika, po technologię przetwarzania i problemy sprzętowe. Są one na pewno ważne, a co istotne — nie należą do utopijnych. W obecnych warunkach są również możliwe do zrealizowania. Nikt przecież nie domagał się natychmiastowej instalacji komputerów IV generacji. Odwrotnie — najwięcej uwagi pochłaniał problem jak w obiektywnych warunkach zwiększyć efektywność obiektywnie możliwych systemów. A wniosek generalny: „Organizacyjne przygotowanie przedsiębiorstwa do automatyzacji jakiejś sfery gospodarowania ma bardzo istotny wpływ na efektywność zastosowań informatycznych". Ze swej strony zapytajmy czy nie najistotniejszy?...

Krystyn BERNATOWICZ

Informatyka w ochronie środowiska

29 czerwca br. odbyła się w Płocku pierwsza sesja na temat „Informatyka w zagadnieniach ochrony środowiska". Sesję zorganizował płocki Zakład nr 3 Ośrodka Badawczo-Rozwojowego Informatyki Gospodarki Terenowej i Ochrony Środowiska. W sesji wzięli udział przedstawiciele Departamentu Ochrony Środowiska Ministerstwa Administracji, Gospodarki Terenowej i Ochrony Środowiska, Ośrodków Badań i Kontroli Środowiska, Instytutu Kształtowania Środowiska, zainteresowanych resortów i wyższych uczelni.

Myślą przewodnią spotkania była ocena dotychczasowych doświadczeń ze współpracy między Ośrodkiem i użytkownikami oraz nakreślenie dalszych kierunków działania w dziedzinie zastosowań informatyki w ochronie środowiska.

W czasie obrad przedstawiono ogólną koncepcję zastosowań informatyki w ochronie środowiska, opracowaną i realizowaną w Zakładzie nr 3 w Płocku, zasady współpracy między użytkownikami systemu i Ośrodkiem oraz zamierzenia na najbliższe lata. W kolejnych wystąpieniach autorzy systemów omówili cel i zakres poszczególnych opracowań oraz wynikające z nich zadania. Zgodnie ze wspomnianą wyżej koncepcją wszystkie prezentowane systemy dotyczyły pierwszego etapu zastosowań informatyki w ochronie środowiska, a szczególnie w dziedzinie ochrony wód; obejmowały więc takie zagadnienia, jak:

- ocena jakości wód powierzchniowych płynących
- ewidencja źródeł zanieczyszczeń wód
- oczyszczalnie ścieków
- naliczanie kar za przekroczenie dopuszczalnych ładunków zanieczyszczeń w ściekach.

Zademonstrowano również działanie wybranych systemów na komputerze ODRA 1305 oraz minikomputerze MERA 305.

Wszyscy uczestnicy spotkania wskazywali na konieczność zorganizowania następnych takich sesji, które uznano za doskonałą formę wymiany doświadczeń. **(ms)**

Popularyzacja opracowań

W dniach 1—2 czerwca br. w Głównym Biurze Studiów i Projektów Energetycznych „Energoprojekt" zorganizowano wystawę publikacji dotyczących zastosowań informatyki w projektowaniu obiektów energetycznych. Przedstawiono bogaty zestaw pozycji literaturowych, obejmujący różne dziedziny i metody automatyzacji projektowania, ze szczególnym uwzględnieniem opracowań wykonanych w Ośrodku Informatyki „Energoprojektu".

Celem ekspozycji było zwiększenie wykorzystania oprogramowania już wykonanego i eksploatowanego, a jednocześnie zainteresowanie środowiska inżynierskiego formułowaniem nowych ciekawych zadań obliczeniowych.

Wybór wydawnictw dokonany był z myślą o inżynierach projektujących elektrownie cieplne, wodne i jądrowe, elektrociepłownie i sieci ciepłownicze oraz stacje i linie elektroenergetyczne. Inżynierom, którzy zetknęli się już z nowoczesną techniką obliczeniową i wykorzystują możliwości ośrodka obliczeniowego w codziennej praktyce projektowej, przedstawiono materiały rozszerzające wiadomości o metodach i językach programowania oraz sprzęcie komputerowym, a także katalogi i informatory o dostępnym w kraju oprogramowaniu inżynierskim. Inżynierom z mniejszym doświadczeniem w wykorzystywaniu elektronicznej techniki obliczeniowej zaprezentowano dokumentację blisko 80 programów eksploatowanych na własnym minikomputerze NOVA-840 i obejmujących takie dziedziny projektowania, jak technologia, budownictwo, instalacje, hydrotechnika, ochrona środowiska, energetyka jądrowa, przesył i rozdział energii elektrycznej, przetwarzanie danych.

Elementem zwiększającym atrakcyjność wystawy była ekspozycja sprzętu.

Według opinii zbieranych na bieżąco od zwiedzających, wystawa spełniła całkowicie założone przez organizatorów zadania, przyczyniając się dzięki dobrze wyeksponowanym przykładom zrealizowanych zastosowań do widocznego wzrostu zainteresowania środowiska inżynierskiego wykorzystywaniem komputerów w projektowaniu.

Zachęcamy ośrodki obliczeniowe działające przy innych biurach projektów do naśladowania inicjatywy „Energoprojektu". **(J. Chaba)**

Kompilator PASCALA w konwersacyjnym systemie CMS

W związku z rosnącymi wymaganiami użytkowników komputerów, dotyczącymi poprawy walorów funkcjonalnych oprogramowania, obserwować można obecnie szczególnie duże zapotrzebowanie na środki programowe umożliwiające pracę w trybie konwersacyjnym.

Zapewnienie użytkownikom języka PASCALA możliwości interakcyjnego uruchamiania programów to główne założenie projektu konwersacyjnego kompilatora tego języka, opracowywanego w Instytucie Maszyn Matematycznych „MERA" dla maszyn Jednolitego Systemu oraz IBM 360/370.

Z punktu widzenia użytkownika nowy kompilator stanowić będzie rozszerzenie istniejącej już wersji wsadowej na maszynach wspomnianych wyżej systemów.

Programy napisane dla tej wersji będą akceptowane przez nowy kompilator, dając identyczne wyniki wykonania.

W ramach prac studialnych, będących etapem przygotowawczym do realizacji kompilatora konwersacyjnego, wykonano włączenie kompilatora wsadowego do systemu CMS działającego w systemie IBM VM/370.

Użytkownik systemu CMS, kontaktujący się z komputerem poprzez terminal, ma do dyspozycji własną maszynę wirtualną, która działa niezależnie od pracujących równocześnie maszyn innych użytkowników. Oprócz szeregu udogodnień dostępnych przy wsadowym trybie pracy, maszyna wirtualna zapewnia nowe możliwości, takie jak interakcyjne uruchamianie programów, napisanych w językach ASSEMBLER, FORTRAN, PL/1.

W przypadku programów w języku PASCAL podobnych możliwości dostarcza kompilator przeniesiony do systemu CMS z systemu OS. Przeniesienia dokonano w oparciu o zasady wykorzystywania zbiorów OS w systemie CMS. Pewne problemy wynikły tu z różnic w mechanizmach łączenia programów w obu systemach.

Przy kompilowaniu programu źródłowego oraz wykonywaniu programu wynikowego w systemie CMS użytkownik ma do dyspozycji szereg środków diagnostycznych i wydawniczych, do których należą programy: EDITOR, DEBUGER, TRACER. Pozwalają one śledzić oraz ingerować w wykonanie programu przy użyciu adresów rzeczywistych. Symboliczne odwoływanie się do składowych programu źródłowego możliwe będzie jednak dopiero w opracowywanej wersji kompilatora konwersacyjnego. Zakłada się, że wersja ta będzie porównywalna z kompilatorami języków PL/1 i FORTRAN, zrealizowanymi przez firmę IBM.

Przy przenoszeniu kompilatora utrzymane zostały wymagania dotyczące wygodnych dla użytkownika zasad działania kompilacji, odporności na błędy oraz ich diagnostyki.

Oprócz dostępnych w systemie OS metod wprowadzania programu źródłowego z kart dziurkowanych, taśmy magnetycznej lub dysku istnieje możliwość wprowadzenia programu źródłowego bezpośrednio z klawiatury terminala bądź za pomocą programu EDITOR. Program ten stosuje się również do poprawiania lub modyfikacji programu źródłowego przechowywanego na dysku w postaci wymaganej przez system CMS. Zasady wywoływania kompilatora zgodne są z konwencjami systemu CMS.

W wyniku działania kompilatora otrzymuje się informacje diagnostyczne przesyłane na ekran terminala, a w przypadku podania odpowiednich opcji, teksty programu źródłowego i programu wynikowego, przechowywane są na dysku. Po zakończeniu kompilacji tekst programu źródłowego może być wypisany na drukarce lub wyświetlony na ekranie terminala.

Przed przejściem do wykonania programu wynikowego programista definiuje na terminalu zbiory odpowiadające plikom PASCALA, z których program ten będzie korzystać.

W przypadku nieprawidłowego zakończenia programu system CMS zapewnia użytkownikowi środki diagnostyczne, do których należy program do testowania interakcyjnego (DEBUGER) oraz program śledzący wykonanie programu wynikowego (TRACER).

Program testujący umożliwia programiście ingerencję w wykonywanie programu wynikowego poprzez:

- ustawianie i usuwanie punktów przerwań (po dojściu do punktu przerwania działanie programu zostaje zawieszone i następuje przejście do trybu konwersacyjnego
- przeglądanie i zmienianie wartości zmiennych programu
- wznawianie wykonywania programu od podanego punktu.

Za pomocą programu śledzącego można otrzymać informacje o skokach w wykonywanym programie, przerwaniach związanych z instrukcjami wejścia i wyjścia itp. Informacje te wyświetlane są dynamicznie na ekranie terminala.

Korzystanie z programów diagnostycznych systemu CMS wymaga posługiwania się rzeczywistymi adresami fraz i zmiennych występujących w programie źródłowym. Informacje te można uzyskać ze skorowidza odwołań oraz z tablicy adresów względnych, tworzonych przez kompilator po podaniu odpowiednich opcji. Pożądane są również elementarne wiadomości o wewnętrznej reprezentacji zmiennych, które znaleźć można w publikacji „PASCAL OS/JS Przewodnik programisty".

Wydaje się, że omówiona wyżej wersja wsadowego kompilatora języka PASCAL działająca w konwersacyjnym systemie CMS w dużym stopniu zaspokoi potrzeby użytkowników przez okres realizacji konwersacyjnej wersji tego kompilatora.

Zofia DZIENISZEWSKA
Grażyna FLEJTA
Instytut Maszyn Matematycznych „MERA"
Warszawa

PRZEGLĄD TELEKOMUNIKACYJNY

— naukowo-techniczny miesięcznik Stowarzyszenia Elektryków Polskich, wydawany przez Wydawnictwa Czasopism Technicznych NOT —

obchodzi złoty jubileusz.

PRZEGLĄD TELEKOMUNIKACYJNY to jedyne czasopismo w Polsce, które prezentuje problematykę dotyczącą takich dziedzin gospodarki narodowej, jak:

- telefonia, telegrafia i teleinformatyka
- radiokomunikacja, radionawigacja i radiolokacja
- radiofonia i telewizja studyjna, nadawcza i odbiorcza
- technika audiowizualna i elektroakustyka
- elektroniczna technika pomiarowa.

PRZEGLĄD TELEKOMUNIKACYJNY doskonale spełnia funkcje propagatora najnowszych rozwiązań światowych, a jednocześnie jest wyrazicielem polskiej myśli naukowo-technicznej w dziedzinie telekomunikacji oraz trybuną wymiany poglądów teleelektryków na aktualne problemy rozwoju nauki, techniki, produkcji i eksploatacji w tej dziedzinie.

Gratulujemy dotychczasowych osiągnięć i życzymy równie udanych następnych roczników!

Usprawnienie programu sprowadzającego inne programy do pamięci operacyjnej w systemie ODRA 1300

Program TAPE jest najczęściej wykonywanym programem w konfiguracjach maszyn cyfrowych serii ODRA 1300 bez dysków magnetycznych i pracujących pod bezpośrednią kontrolą programu EXECUTIVE. TAPE jest pierwszym programem na każdej bibliotecznej taśmie magnetycznej i służy do sprowadzania do pamięci operacyjnej pozostałych programów znajdujących się na tej taśmie. Z tych powodów każde usprawnienie pracy programu TAPE przyczynia się do lepszego wykorzystania czasu pracy maszyn cyfrowych serii ODRA 1300.

W wersji dotychczasowej program TAPE podczas poszukiwania określonego programu na taśmie magnetycznej czyta kolejne zapisane na niej bloki aż do momentu napotkania bloku żądań poszukiwanego programu. Następnie taśma magnetyczna jest cofana i ustawiana przed tym blokiem, odtwarzana jest pierwotna zawartość akumulatorów, tj. ich stan przed wprowadzeniem programu TAPE do pamięci operacyjnej; na monitor zostaje wyprowadzony komunikat o znalezieniu żądanego programu, po czym program TAPE kasuje się automatycznie.

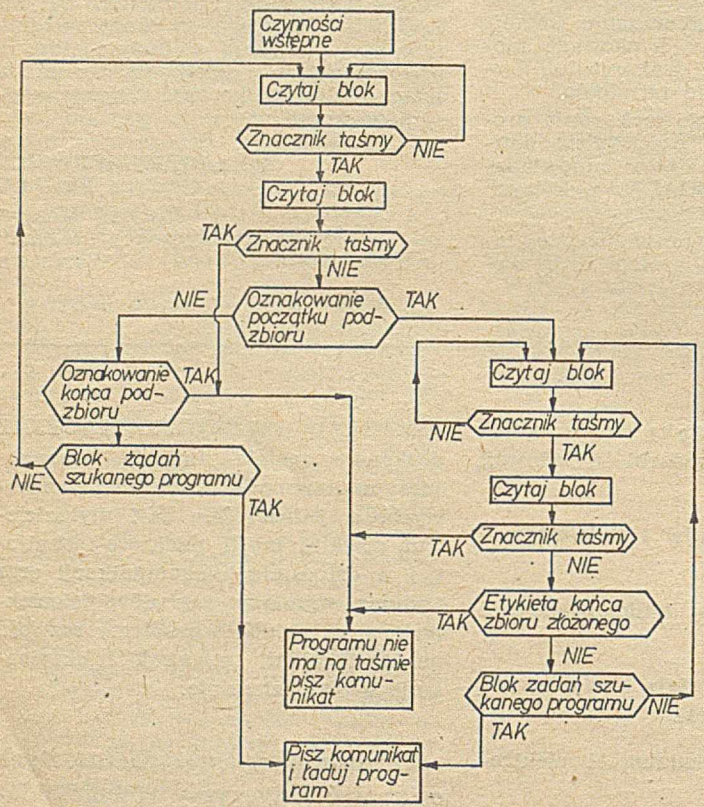
Komunikat o znalezieniu programu interpretowany jest przez EXECUTIVE jako polecenie operatorskie załadowania tego programu do pamięci operacyjnej. Jeśli żądany program nie został znaleziony na taśmie magnetycznej, program TAPE wyprowadza odpowiedni komunikat na monitorze po przeszukaniu całej taśmy magnetycznej (do znacznika końca zbioru). Schemat blokowy zasadniczej części programu TAPE, tj. algorytm przeszukiwania taśmy magnetycznej aż do napotkania bloku żądań poszukiwanego programu, pokazany jest na rys. 1.

Modyfikacja programu TAPE polega na zmianie algorytmu przeszukiwania bibliotecznej taśmy magnetycznej. Dokonana zmiana algortymu oparta jest na obecności znacznika taśmy (ang. *tape mark*) bezpośrednio przed blokiem żądań każdego programu umieszczonego na taśmie bibliotecznej. Zmodyfikowany program TAPE zawiera w swym obszarze stałych zbiór informacji o programach zapisanych na danej taśmie bibliotecznej, zwany dalej katalogiem. W uruchomionej wersji katalog zajmuje 385 słów maszyny. Pierwsze z nich zawiera liczbę programów (prócz programu TAPE) umieszczonych na danej taśmie bibliotecznej, a następnie podzielone są na 128 grup po 3 słowa. Każda grupa zawiera następujące informacje o kolejnym programie na taśmie bibliotecznej:

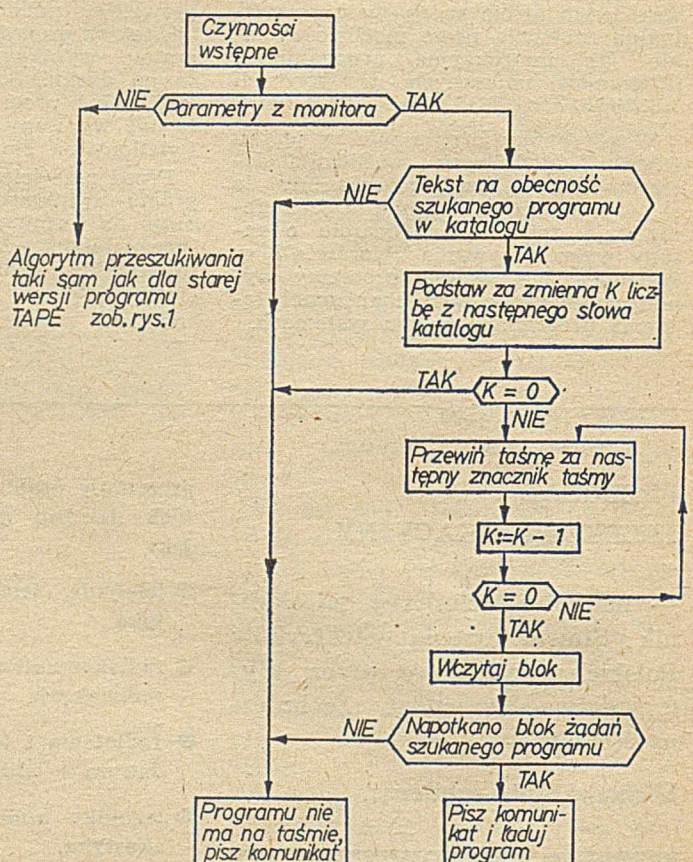
— słowo 1: nazwa programu
— słowo 2: liczba znaczników taśmy poprzedzających dany program, zmniejszona o jeden
— słowo 3: określenie wersji programu.

Zmodyfikowany program TAPE działa według następującego algorytmu (patrz schemat blokowy na rys. 2). W katalogu znajdowana jest grupa 3 słów odpowiadających poszukiwanemu programowi. Następnie, wykorzystując rozkaz przewinięcia taśmy magnetycznej poza znacznik taśmy i informacje uzyskane z katalogu, głowica czytająca ustawiana jest przed blokiem żądań poszukiwanego programu i przeprowadzany jest test zgodności nazw poszukiwanego i znalezionego programu. W przypadku pozytywnego wyniku testu dalsze postępowanie jest identyczne jak w dotychczasowej wersji programu TAPE. Negatywny wynik testu, spowodowany np. błędem taśmy magnetycznej, jest interpretowany jako nieobecność poszukiwanego programu na taśmie bibliotecznej. Nieobecność nazwy poszukiwanego programu w katalogu powoduje wyprowadzenie na monitor odpowiedniego komunikatu bez przeszukiwania taśmy magnetycznej.

W zmodyfikowanej wersji programu nowy algorytm przeszukiwania taśmy magnetycznej używany jest tylko w tym przypadku, gdy komunikat operatorski żąda sprowadzenia określonego programu z taśmy, na której pierwszym programem jest zmodyfikowany program TAPE (wejście do programu od słowa 22). W pozostałych przypadkach (tj. gdy program TAPE jest sterowany dodatkowymi parametrami wczytanymi z kart lub taśmy papierowej i uruchomiony od słowa 20 lub 21, przeszukuje inną taśmę magnetyczną niż ta, z której został załadowany do pamięci) algorytm przeszukiwania taśmy pozostaje niezmie-



Rys. 1. Schemat blokowy dotychczasowej wersji programu TAPE



Rys. 2. Schemat blokowy zmodyfikowanego programu TAPE

niony. Używając wejścia 20 lub 21 można też sprowadzić do pamięci program znajdujący się na tej samej taśmie magnetycznej, co zmodyfikowany program TAPE, ale nie umieszczony w katalogu. Wprowadzone zmiany spowodowały zwiększenie objętości programu TAPE po załadowaniu. W zmodyfikowanej wersji zajmuje on 1344 słów, dotychczasowa zaś wersja zajmowała 512 słów. Podczas modyfikacji nie zmieniono postaci parametrów ani komunikatów wyprowadzanych na monitor. Tak więc operatorska obsługa zmodyfikowanego programu TAPE jest taka sama, jak obsługa starej wersji tego programu.

Obecność katalogu w zmodyfikowanym programie TAPE wymaga zmiany procedury tworzenia i aktualizacji bibliotecznych taśm magnetycznych. Każda zmiana zawartości taśmy bibliotecznej wymaga bowiem zaktualizowania katalogu. Zmieniona procedura tworzenia lub aktualizacji taśmy bibliotecznej ze zmodyfikowanym programem TAPE jako programem sprowadzającym, składa się z 3 kolejnych faz:

1) utworzenia lub aktualizacji taśmy bibliotecznej
2) zaktualizowania katalogu i umieszczenia go wewnątrz zmodyfikowanego programu TAPE
3) umieszczenia zmodyfikowanego programu TAPE na początku nowej taśmy bibliotecznej.

Faza pierwsza i trzecia mogą być wykonane za pomocą programu aktualizacji bibliotek XPMV. Do wykonania fazy drugiej służy specjalnie napisany program CATA.

Konieczność wykonania czasochłonnej procedury aktualizacji taśm bibliotecznych (wykonywanej jednak stosunkowo rzadko) jest w pełni rekompensowana przez zwiększenie szybkości działania zmodyfikowanego programu TAPE w porównaniu z dotychczasową wersją.

W celu zmierzenia efektów modyfikacji programu TAPE utworzono dwie taśmy biblioteczne różniące się tylko programem sprowadzającym. Na jednej z nich była dotychczasowa wersja programu TAPE, a na drugiej zmodyfikowany program TAPE. Zmierzono czas ładowania i działania programów sprowadzających od momentu napisania komunikatu na monitorze przez operatora do momentu wypisania przez program sprowadzający komunikatu na monitorze o znalezieniu szukanego programu. Doświadczenie wykazało, że modyfikacja programu TAPE przyspieszyła jego działanie o 34% w porównaniu z dotychczasową wersją.

Szczegółowy opis działania zmodyfikowanego programu TAPE i programu CATA aktualizującego katalog można otrzymać od autorów niniejszej pracy. Kopie obu programów (w wersji źródłowej i binarnej) przesyłamy bezpłatnie na taśmie magnetycznej zainteresowanego użytkownika (prosimy jedynie o podanie gęstości zapisu).

Piotr GIZBERT-STUDNICKI
Instytut Fizyki UJ i CYFRONET Kraków
Jerzy PTAK
Instytut Informatyki UJ Kraków

O wykorzystaniu symbolu jednostki miary

Od niedawna obowiązuje nowy wykaz jednostek miar, opracowany przez Ośrodek Badawczo-Rozwojowy Systemu Państwowej Iinformacji Statystycznej na potrzeby Kodu Towarowo-Materiałowego (KTM). Wykaz ten zawiera nazwę jednostki miary, oznaczenie literowe i oznaczenie cyfrywe. Wykaz podzielony jest na dwie części. Pierwsza to jednostki podstawowe, a druga jednostki umowne. Każda z tych części dzieli się na grupy. W skład grupy wchodzą jednostki dotyczące tej samej wielkości.

Jest to już druga symbolika jednostek miar, która jednak zarówno jak poprzednia z Systematycznego Wykazu Wyrobów (SWW) nie pozwala na proste przeliczanie ilości podanej w jednej jednostce na ilość w innej jednostce (dotyczy przeliczania jednostek należących do jednej grupy).

I tak w SWW przeliczenie kg na Mg musiało być poprzedzone badaniem, czy z jednej strony symbol jest równy 33 a z drugiej 34. Wówczas dopiero ilość dzielono przez 1000. W każdym przypadku przeliczenie ilości z jednej na inną jednostkę miary musiało być poprzedzone innym, indywidualnym badaniem. Podobnie jest obecnie w KTM: oznaczenie cyfrowe jednostki miary (aktualnie składające się z trzech cyfr) służy więc jedynie do identyfikacji jednostek. W tym miejscu nasuwa się pytanie. Czy nie można tak dobrać symboliki, aby przeliczanie ilości z jednej jednostki miary na inną było proste?

Dobór symboliki może być następujący. Cały zakres symbolu od 000 do 999 dzielimy na grupy jednakowo liczne. Liczność grup oznaczmy przez LG. Otrzymamy wówczas 1000/LG grup jednostek. Jednostce najmniejszej w danej grupie przydzielamy najmniejszy symbol z zakresu grupy. Kolejne symbole z zakresu grupy przydzielamy jednostkom odpowiednio 10, 100, 1000 itd. razy większym od jednostki najmniejszej.

Przykład.

Dla LG = 10 otrzymamy 1000/10 = 100 możliwych grup jednostek. Jedna z grup np. jednostek pieniężnych (stanowiąca — powiedzmy — grupę) wygląda następująco:

symbol jednostki	nazwa jednostki
001	1 grosz
002	10 groszy
003	1 zł
004	10 zł
005	100 zł
006	1 tys. zł
007	10 tys. zł
008	100 tys. zł
009	1 mln zł

Grupa jednostek liczebności, stanowiąca np. jedenastą grupę, przedstawiałaby się następująco:

symbol jednostki	nazwa jednostki
100	1 sztuka
101	10 sztuk
102	100 sztuk
103	1 tys. szt.
104	10 tys. szt.
105	100 tys. szt.
106	1 mln szt.
107	10 mln szt.
108	100 mln szt.
109	1 mld szt.

Podobnie można opisać pozostałe grupy. W przypadku gdyby któraś z jednostek nie była wykorzystana, dany symbol nie występowałby w wykazie. Dla symboliki określonej według tej metody przeliczanie ilości jest bardzo proste.

Oznaczmy ilość podaną w jednostce JM1 przez IL1 i jednostkę, na którą przeliczamy, przez JM2.

Schemat przeliczania przedstawia się następująco:

- część całkowita z podzielenia symbolu jednostki przez liczność grup (LG) informuje o tym, do której grupy dana jednostka należy
- gdy obie jednostki należą do tej samej grupy, współczynnik przeliczeniowy wyliczamy podnosząc liczbę 10 do potęgi równej różnicy między symbolami tych jednostek. Jak widać jest to operacja prosta i nie zależy od JM1 ani JM2.

Przeliczanie dla jednostek należących do różnych grup, np. przeliczenie m^3 na kg czy sztuk na kg ma charakter indywidualny dla danego materiału i w sposób ogólny nie może być zdefiniowane.

Niemniej i w tym wypadku przy tak zbudowanej symbolice podanie współczynnika do przeliczenia sztuk na kg ułatwiłoby przeliczanie z jednostek pochodnych jednostce „sztuka" na jednostki pochodne jednostce „kg".

Symbolika taka eliminowałaby szereg ograniczeń nakładanych obecnie na postać danych w systemach EPD, szczególnie w systemach bazujących na danych zbieranych z różnych źródeł.

Piotr OMIECIŃSKI
ZWCH „Chemitex-Celwiskoza"
Jelenia Góra

Przemysł informatyczny NRD na Wiosennych Targach Lipskich 1978

W ciągu kilku ostatnich lat obserwowaliśmy w NRD konsekwentne poczynania organizacyjne przemysłu informatycznego oraz przemysłu maszyn biurowych, zmierzające do usprawnienia wzajemnej koordynacji i specjalizacji oraz unowocześnienia i dostosowania produkowanego sprzętu do aktualnych trendów rozwojowych.

Przypomnijmy, że produkcja sprzętu informatycznego w NRD była rozmieszczona w dwóch kombinatach przemysłowych, z których jeden (ZENTRONIK) specjalizował się w wytwarzaniu urządzeń do przygotowywania danych i maszyn biurowych, natomiast drugi (ROBOTRON) produkował komputery i ich urządzenia peryferyjne.

Taka struktura mogła być uzasadniona tylko do czasu, gdy istniał wyraźny podział techniczno-organizacyjny, uzasadniający oddzielne traktowanie problemów sporządzania nośników danych i przetwarzania danych. Dodatkowym czynnikiem był fakt, że znaczna część urządzeń do sporządzania nośników danych powstała jako konstrukcyjne rozwinięcie konwencjonalnych maszyn biurowych uzupełnionych dziurkarkami taśm lub kart. Wraz z postępującym zastępowaniem w tych maszynach zespołów elektromechanicznych układami elektronicznymi, w obu kombinatach powstała konieczność koordynacji prac naukowo-badawczych i produkcji oraz daleko idącej unifikacji elementów (modułów) elektronicznych.

Jest faktem, że w większości systemów informatycznych zatarła się już istniejąca dotąd granica między przygotowywaniem danych a ich przetwarzaniem. Obecna faza przygotowania danych ściśle wiąże się z ich szybkim wstępnym przetwarzaniem na potrzeby operatywnego zarządzania, a dopiero później uzyskane stąd wyniki opracowywane są w następnym etapie przetwarzania dla celów planowania, sprawozdawczości rozliczeń, analiz itp. Doprowadziło to do szczególnie szybkiego rozwoju i rozpowszechnienia urządzeń tzw. średniej automatyzacji. W tej sytuacji dotychczasowy profil produkcyjny przemysłu informatycznego NRD nie odpowiadał w pełni nowoczesnym założeniom organizacyjnym systemów informatycznych.

W związku z powyższym w styczniu br. kombinat ROBOTRON objął całość produkcji sprzętu informatycznego i maszyn biurowych NRD. W ramach tego kombinatu działa obecnie 19 zakładów przemysłowych i ośrodków badawczych oraz własne przedsiębiorstwo eksportowo-importowe o łącznym zatrudnieniu ok. 70 000 pracowników, co wysunęło go na pierwsze miejsce wśród największych organizacji przemysłowych NRD. Oczekuje się, że koncentracja produkcji sprzętu informatycznego usprawni przede wszystkim koordynację w wytwarzaniu i stosowaniu elementów i modułów elektronicznych, pozwoli na lepsze wykorzystanie zaplecza naukowo-badawczego i możliwości produkcyjnych oraz usprawnienie zarządzania tą gałęzią gospodarki.

Dotychczasowa suwerenność niektórych zakładów stanowiła źródło pewnego rodzaju działań konkurencyjnych, rozpraszających potencjał produkcyjny i zmniejszających tempo modernizacji wyrobów. Oczywiście wypada poczekać rok lub dwa zanim nowa organizacja okrzepnie i dopiero wówczas będzie można dokonać pierwszych ocen. Jednakże już dzisiaj wypada zwrócić uwagę, że zarządzanie tak wielką organizacją gospodarczą, o bardzo zróżnicowanym asortymencie wyrobów, może ujawnić nowe, nieprzewidziane okoliczności. Z tego chociażby względu obserwacja takich poczynań może być dla nas bardzo pożyteczna.

Ekspozycja przemysłu sprzętu informatycznego na tegorocznych wiosennych Targach Lipskich jest pierwszą organizowaną przez nowy kombinat. Zawiera już ona cechy świadczące o nowym kierunku zamierzonego rozwoju. M.in. całkowicie zrezygnowano z prezentowania maszyn mechanicznych (z wyjątkiem maszyn do pisania), co w konfrontacji z rokiem ubiegłym stanowi zasadniczą zmianę. Równocześnie z utrzymanych jeszcze w produkcji urządzeń elektronicznych poprzedniej oferty, wyselekcjonowano do zaprezentowania na Targach jedynie te, które wykazują cechy nowoczesności, zarówno konstrukcyjne, jak i w zastosowaniu. W większości przypadków prezentowane modele wywodzą się z zestawu znormalizowanych modułów, z których tworzy się urządzenia o zróżnicowanych cechach eksploatacyjnych, dostosowanych do różnych zastosowań. Jest to niewątpliwie zjawisko pozytywne, prowadzące poprzez standaryzację konstrukcji i technologii do znacznej poprawy warunków eksploatacji technicznej.

Nie można na razie mówić o zdecydowanych nowościach, każde jednak z prezentowanych urządzeń miało nowe elementy eksploatacyjne, dopracowane w okresie ostatniego roku.

Na szczególną uwagę zasługują następujące urządzenia — wymienione w kolejności uwzględniającej masowość zapotrzebowania naszych odbiorców.

Elektroniczna maszyna do księgowania daro 1750

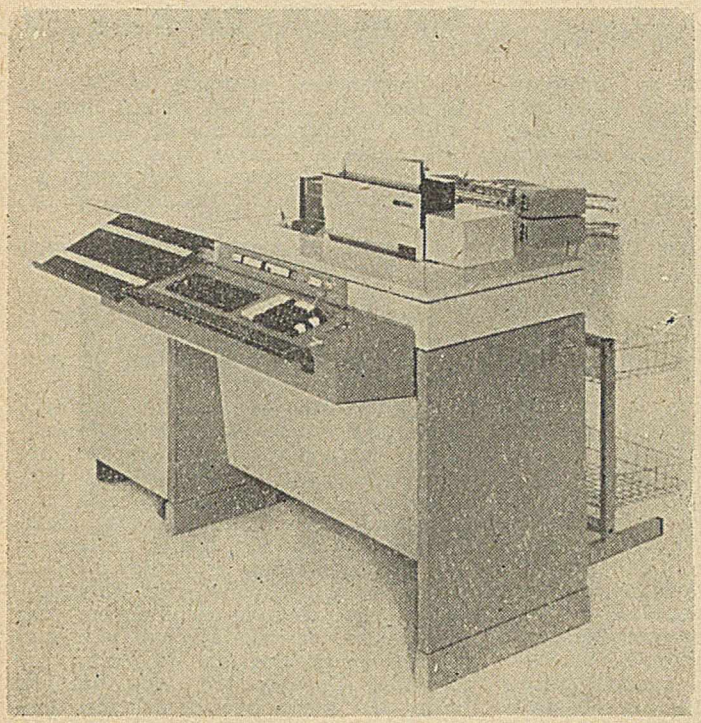
Elektroniczna maszyna do księgowania daro 1750 pomyślana jest jako urządzenie zastępujące dotychczas najpopularniejszą wśród naszych użytkowników, mechaniczną maszynę do księgowania Ascota kl. 170. Daro 1750 jest urządzeniem opartym na zminiaturyzowanych obwodach scalonych, przeznaczonym przede wszystkim do wykonywania zadań z zakresu rachunkowości konwencjonalnej. Zarówno elektroniczna konstrukcja, jak i wyposażenie w dodatkowe zespoły, znacznie zwiększają zakres zastosowania oraz wydajność tej maszyny w porównaniu do wspomnianej wersji mechanicznej.

Zestaw standardowy daro 1750 obejmuje jednostkę centralną z pamięcią ferrytową o pojemności 8 K bajtów. Pamięć przeznaczona jest dla programów oraz danych i może być dowolnie programowana. Maszyna w wersji standardowej wyposażona jest w 2 kanały wejściowe i 2 kanały wyjściowe.

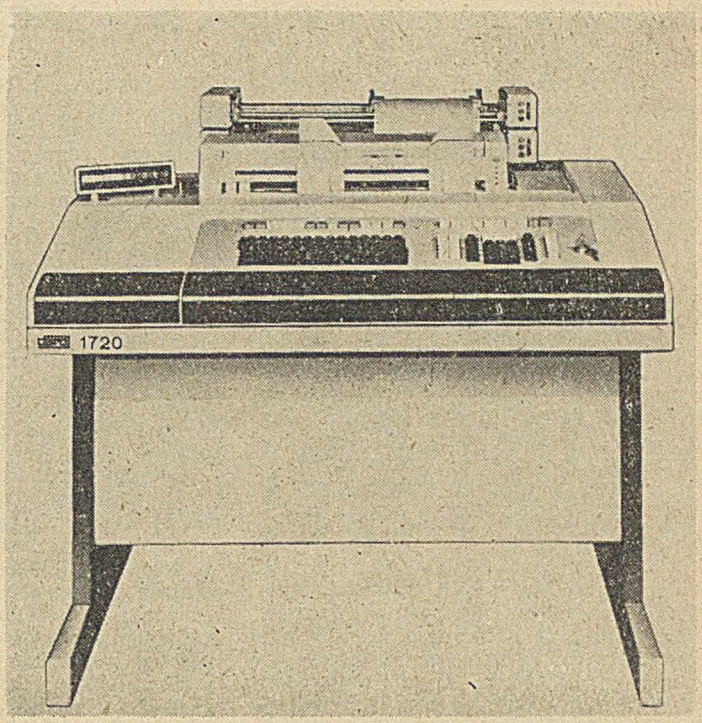
Wprowadzanie programów może odbywać się za pomocą nośnika magnetycznego lub przez klawiaturę. Maszyna jest wyposażona w klawiaturę alfanumeryczną oraz oddzielną klawiaturę dziesiętną i funkcyjną. Wyprowadzenie wyników odbywa się za pomocą drukarki mozaikowej o szybkości 100 znaków/s.

Maszyna może być wykorzystywana w rachunkowości z zastosowaniem konwencjonalnych kart kontowych, jednak efektywność jej pracy znacznie zwiększa zastosowanie przystawki do magnetycznych kart kontowych oraz urządzenia do formularzy ciągłych (leporello).

Do maszyn daro 1750 można również dołączyć pamięć na dyskach elastycznych o pojemności 256 K bajtów, spełniających zarówno rolę pamięci zewnętrznej, jak i nośnika informacji dla dalszego przetwarzania na komputerze.



Elektroniczna maszyna do księgowania daro 1750



Elektroniczny automat obrachunkowy daro 1720

Elektroniczny automat obrachunkowy daro 1720

Elektroniczny automat obrachunkowy daro 1720 jest nowoczesną konstrukcją, opartą na zastosowaniu obwodów scalonych typu PROM. Maszyna jest programowana za pomocą dwóch kaset programowych, zawierających po cztery elementy PROM (łącznie 512 rozkazów 16-bitowych). Łatwo wymienne kasety pozwalają na szybką zmianę programów, a więc na elastyczne stosowanie maszyny.

Maszyna daro 1720 jest oferowana w różnych wariantach wyposażenia w urządzenia dodatkowe i peryferyjne. W porównaniu do daro 1750 jest ona dostosowana do fakturowania i prac rozliczeniowych.

Urządzenie do przygotowywania nośników danych daro 1370

Urządzenie daro 1370 służy do programowanego przygotowania danych na kasetowych taśmach magnetycznych. Jest ono wyposażone w klawiaturę alfanumeryczną i numeryczną oraz monitor ekranowy lub drukarkę. Nośnikiem danych może być również dysk elastyczny lub taśma dziurkowana.

Do konwersji uzyskanej z daro 1370 taśmy kasetowej na standardową półcalową taśmę komputerową oferowany jest programowany konwerter daro 1255, wyposażony w pamięć operacyjną o pojemności 4 K bajtów.

Wielostanowiskowy system rejestracji danych ROBOTRON 4230

Całkowicie nową propozycją kombinatu był wielostanowiskowy system rejestracji danych ROBOTRON 4230, podobny w założeniu do wieloklawiaturowego systemu MERA-SEECHECK.

Komputer ROBOTRON EC 1055

Komputer ROBOTRON EC 1055 jest niemieckim reprezentantem następnej generacji maszyn Jednolitego Systemu zwanej „drugim pokoleniem" lub „generacją 3 1/2". Zaliczany jest do grupy komputerów średniej wielkości wg współczesnych kryteriów klasyfikacyjnych. W modelu tym zastosowano pamięć wirtualną, rozszerzoną listę rozkazów, dwubajtowy interfejs, zwiększoną dokładność wykonywania operacji zmiennoprzecinkowych, kanały multipleksorowe oraz ulepszony system operacyjny.

Komputer EC 1055 został zaprezentowany na Targach w następującej konfiguracji: jednostka centralna EC 2655, jednostka obsługi operatorskiej i technicznej z monitorem ekranowym EC 7069, drukarka wierszowa EC 7033, pamięć taśmowa EC 5017, mikrofilmowe urządzenie wyjściowe EC 7602 oraz pamięć dyskowa EC 5066 o pojemności 100 M bajtów.

Mimo że wystawiany egzemplarz stanowi niewątpliwie obiekt prototypowy, akcentowano fakt istnienia pełnego oprogramowania problemowego, co świadczy o pełnej programowej kompatybilności z produkowanym dotąd modelem EC 1040, z którego przeniesiono niewątpliwie całość dobrze już sprawdzonego w eksploatacji oprogramowania.

Optyczny czytnik znaków kreskowych daro 1375

Jest to urządzenie odczytujące optycznie oraz interpretujące kreski nanoszone w określonych miejscach specjalnie zaprojektowanego formularza. W wyniku automatycznego odczytu danych następuje ich zapisanie na standardowej taśmie magnetycznej, służącej do dalszego przetwarzania na komputerze.

Jest to jak wiadomo najwcześniejsze i najprostsze rozwiązanie automatycznego odczytu dokumentów, dość powszechnie jeszcze stosowane na świecie ze względu na stosunkowo niski koszt, lecz ograniczone głównie do zastosowań w handlu (zamówienia standardowego asortymentu towarów) lub przy inkasie opłat za zużycie energii. Jest to niwątpliwie pierwsze urządzenie tego typu skonstruowane w krajach RWPG.

W sumie należy uznać, że tegoroczna prezentacja przemysłu sprzętu informatycznego NRD była koncepcyjnie bardziej zwarta niż w latach ubiegłych, potwierdzając przyjęty zgodnie z aktualnymi trendami światowymi kierunek rozwoju tego sprzętu.

(mb)

Głosowe urządzenie wejściowe firmy NEC

Głosowe urządzenie wejściowe stosowano dotychczas raczej rzadko, ale rozwój mikroelektroniki, a w szczególności mikroprocesorów, pozwala już na budowę takich urządzeń, które łączą doskonałe parametry ze stosunkowo niską ceną, co rokuje nadzieje na ich szersze rozpowszechnienie.

Głosowe urządzenie wejściowe, opracowane ostatnio przez japońską firmę NIPPON ELECTRIC CORP (NEC), może rozpoznawać mowę w zakresie 120 słów, przy czym słowa mogą być wypowiadane w blokach liczących do 5 słów, z pauzą między blokami. Dotychczas większość urządzeń tego typu mogła rozpoznawać tylko pojedyncze słowa, a nie ich całe grupy — przy czym zestaw był z reguły ograniczony do kilku, co najwyżej kilkunastu słów.

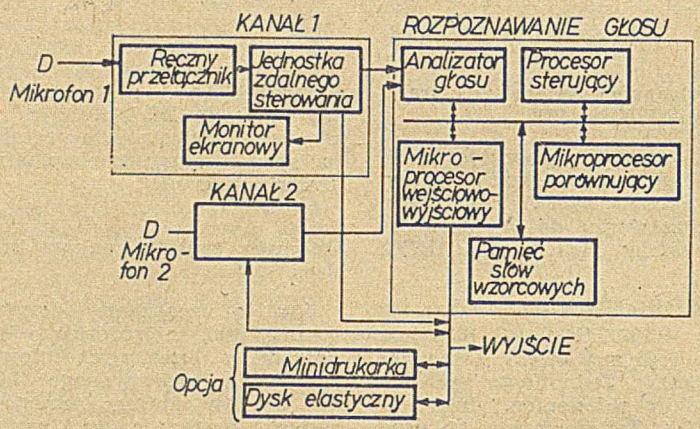
Urządzenie firmy NEC zawiera: analizator głosu, pamięć słów wzorcowych oraz trzy mikroprocesory — porównujący, wejściowo/wyjściowy i sterujący; połączone są one w sposób przedstawiony na załączonym schemacie blokowym.

Bardzo szybki 16-bitowy mikroprocesor sterujący, złożony z 4-bitowych układów typu 2901, wykonuje funkcje sterujące i decyzyjne, np. przetwarza informacje związane z początkiem i końcem słowa lub grupy słów, które są dostarczane z analizatora głosu.

Słowa, które mają być rozpoznawane, wprowadza się do urządzenia za pomocą czytnika taśmy dziurkowanej. Dzięki monitorowi ekranowemu operator odczytuje od razu każde słowo, którego odpowiednik głosowy jest aktualnie zapisywany w pamięci. Dla zapisania każdego słowa w dynamicznej pamięci o bezpośrednim dostępie zużywa się około 200 bajtów.

Podczas prezentacji urządzenia w Tokio stosowano alfabet łaciński, ale można stosować także inne alfabety.

Standardowa konfiguracja obejmuje 2 niezależne kanały wejściowe, pozwalające na rozpoznawanie dowolnej sekwencji ze 120 słów. Możliwości urządzenia mogą być rozszerzone przez dołączenie pamięci dyskowej, przechowującej słowa, celem ich późniejszej obróbki w trybie *off-line* — wtedy urządzenie może obsługiwać aż cztery operatorów jednocześnie; inne urządzenia tego typu obsługują na ogół tylko jednego operatora.



Schemat blokowy głosowego urządzenia wejścia firmy NEC. Zawiera ono bipolarny procesor porównujący, mikroprocesor wejściowo/wyjściowy (8080A) i bipolarny procesor sterujący

Analizator głosu analizuje częstotliwości mowy ludzkiej w zakresie od 120 do 5900 Hz, w 16 oddzielnych kanałach. Procesor porównujący odbieraną mowę z zawartością pamięci słów wzorcowych jest wykonany z bipolarnych układów scalonych TTL, co zapewnia dużą szybkość działania. Jego programowanie jest dynamiczne — może być zmieniane przez procesor sterujący, np. w zakresie modyfikacji skali czasu. Procesor wejściowo/wyjściowy (w urządzeniu wykorzystano własną wersję mikroprocesora 8080 A, produkowanego przez firmę NEC pod nazwą „μcom-8") realizuje sprzężenie pomiędzy głosowym urządzeniem wejścia a jednostkami zdalnego sterowania, które obsługują dwa stanowiska operatorskie. Steruje on także danymi wprowadzanymi z czytnika taśmy dziurkowanej podczas procesu wprowadzania słów wzorcowych do pamięci, a w razie potrzeby może sterować pamięcią dyskową i minidrukarką.

Jeszcze jedną zaletą urządzenia firmy NEC jest około 2-krotnie większa szybkość niż w urządzeniach konkurencyjnych. Rozpoznanie jest realizowane w 300 milisekund po zakończeniu wprowadzania każdego bloku słów, co pozwala na wprowadzanie 60 słów na minutę.

Opisywane głosowe urządzenie wejściowe będzie sprzedawane początkowo tylko w Japonii. Przewiduje się jego zastosowanie w handlowych systemach informatycznych, podczas inwentaryzacji i kontroli, w gospodarce magazynowej itp. Testy wykazują dobre parametry urządzenia, które działa poprawnie nawet przy dużym hałasie. Ponadto firma NEC oferuje na żądanie rozszerzone możliwości rozpoznawania do zestawu 400 słów.

Opr. Z. NAOTYŃSKI
(ELECTRONICS z 13 kwietnia br.)

System VIEWDATA

Ministerstwo łączności w Wielkiej Brytanii (British Post Office) opracowało i wdraża obecnie do eksploatacji rozwiązanie polegające na wykorzystaniu odbiornika telewizyjnego jako terminala w ogólnodostępnym systemie informacji.

Konwersacyjny, usługowy, wspomagany komputerem system informacyjny o nazwie VIEWDATA został opracowany z myślą o tzw. „powszechnym" użytkowniku. Zapewnia on dostęp w trybie *on-line* do bazy danych.

Terminalem użytkownika jest zwykły odbiornik telewizyjny (zarówno dla emisji czarno-białej, jak i kolorowej), specjalnie adaptowany do nowych zadań przez zespolenie ze zbudowanym na układach scalonych modemem i generatorem znaków. Zmodyfikowany w ten sposób odbiornik telewizyjny, który w dalszym ciągu może odbierać również normalny program, jest połączony z telefonem oraz klawiaturą, umożliwiającymi użytkownikowi przekazywanie do komputera zapytań. Dodatkowym wyposażeniem terminala może być urządzenie drukujące odpowiedzi systemu.

Procedura korzystania z systemu polega na tym, że użytkownik wykręca numer telefonu swego lokalnego ośrodka w systemie VIEWDATA, uzyskując dostęp do minikomputera, który może przyjąć wiele takich zgłoszeń równocześnie. Odpowiedzi wyświetlane są w postaci stronic (1 stronica wypełnia powierzchnię ekranu odbiornika telewizyjnego), powiązanych hierarchicznie w postaci 9-poziomowego „drzewa". Obecnie pojemność bazy danych wynosi 98 000 takich stronic.

Wprowadzanie do bazy danych materiałów informacyjnych odbywa się za pomocą urządzenia klawiaturowego lub taśmy magnetycznej. Podstawą systemu jest komputer GEC 4080.

System VIEWDATA nie obsługuje działalności placówek informacji naukowo-technicznej, lecz spełnia funkcje realizowane obecnie przez prasę, kursy szkoleniowe i konwencjonalną pocztę. Na razie system jest dostępny dla użytkowników mieszkających w rejonie Londynu, Birmingham i Norwich. Ostatnio system VIEWDATA zakupiła RFN i rozpoczęła prace wdrożeniowe.

Inną formą konwersacyjnych telewizyjnych systemów informatycznych, noszącą nazwę „Teletext systems", są systemy: CEEFAX (własność BBC) oraz ORACLE (własność ITA). Systemy te posługują się również zmodyfikowanym domowym odbiornikiem telewizyjnym dla przesyłania informacji w formie stronic za pośrednictwem kanałów transmisji telewizyjnej, nie zajętych przez normalny program. Przez naciśnięcie klawisza użytkownik może wybrać jedną ze 100—200 stronic zawartych w systemie. Obecny zakres dostępnych informacji obejmuje komunikaty o programach radiowych i telewizyjnych, prognozy pogody, wiadomości, informacje insruktażowe.

J. ZEMBRZUSKI

Wydruk na papierze to jeszcze nie wynik epd!

Rdzeniem każdego systemu informatycznego jest niewątpliwie proces przetwarzania danych. Powoduje to koncentrację wysiłków zmierzających do najefektywniejszego wykorzystania możliwości organizacyjnych i technicznych komputera i jego urządzeń peryferyjnych. Niejako w cieniu tego podstawowego zadania znajduje się problem przygotowywania danych. Dowodem tego są m.in. ciągle jeszcze prymitywne środki techniczne używane do sporządzania maszynowych nośników informacji. Jest to niewątpliwie jeden z poważnych czynników hamujących sprawność i zmniejszających efektywność działania systemów informatycznych. O problemie tym mówi się jednak i coraz bardziej uwzględnia w założeniach nowo projektowanych systemów. Mówiąc jednakże o pełnej efektywności systemu należy pamiętać również o właściwym wykorzystaniu uzyskanych wyników oraz prezentowaniu ich odbiorcom w formie odpowiednio dostosowanej do bezpośredniego wykorzystania. Ten właśnie problem stanowi jednak najczęściej margines zainteresowań projektantów systemów i w niektórych przypadkach może prowadzić do dezaktualizacji wyników przetwarzania lub wręcz uniemożliwiać praktyczne ich wykorzystywanie.

Podstawowym nośnikiem wyjściowym wyników z komputera jest nadal — i zapewne długo jeszcze pozostanie — powszechnie stosowana w drukarkach ciągła taśma papierowa. Na taśmie tej wypisywane są dokumenty różnych formatów, często umieszczone po kilka obok siebie. Mogą to być również dokumenty sporządzane w kilku egzemplarzach, przez kalkę, np. faktury, wyciągi bankowe itp.

Jednak dopiero dokumenty powstałe z papieru ciągłego po odpowiednim pocięciu i posortowaniu stanowią rzeczywisty użytkowy wynik działania systemu. Od tego, jak szybko i sprawnie zostaną wykonane te operacje, zależy praktyczna efektywność systemu. W stwierdzeniu tym nie ma przesady. Jeżeli na przykład system opracuje dane w ciągu kilku godzin i sporządzi tysiące faktur, zaś rozdzielenie i pocięcie tego materiału trwać będzie jeden lub kilka dni, to paradoks jest oczywisty. Dotyczy to każdej dziedziny, w której dokument z drukarki komputera stanowi podstawę do dalszego działania lub podejmowania operatywnych decyzji.

Obecna praktyka polega w większości przypadków na stosowaniu prymitywnych gilotyn, a często nawet na ręcznym cięciu tabulogramów, szczególnie wtedy, gdy gilotyna nie może usprawnić pracy. W przypadku tabulogramu, w którym w jednym rzędzie znajduje się kilka kolejno uszeregowanych dokumentów, stosowanie zwykłej gilotyny jest trudne, jeżeli w pakiecie trzeba zachować właściwą kolejność dokumentów (np. wg kolejnych numerów). Taki stan może oczywiście powodować w wielu przypadkach całkowitą utratę efektów przyspieszenia i zmniejszenia pracochłonności operacji, uzyskanych w wyniku zastosowania elektronicznego przetwarzania danych.

Poszukiwania zmierzające do usprawnienia metod końcowego opracowywania drukowanych wyników przetwarzania na komputerze poczyniły w ostatnich latach duże postępy. Obecnie stosowane rozwiązania to elektronicznie sterowane urządzenia o dużej wydajności i elastyczności w dostosowywaniu do konkretnych potrzeb użytkowników. Przykładem bardzo efektywnych systemów opracowywania wydruków na formularzach ciągłych, w szeroko pojętym zrozumieniu tego określenia, mogą być maszyny i urządzenia wyspecjalizowanej w tej dziedzinie firmy BÖWE (RFN), które można spotkać w bardzo wielu ośrodkach obliczeniowych.

Podstawowymi operacjami przy końcowym opracowywaniu wyników otrzymanych z drukarek na papierze ciągłym są: dzielenie, rozłączanie i kopertowanie.

Dzielenie

Dokumenty umieszczone na papierowej taśmie ciągłej muszą być zwykle podzielone na żądany format. Ponadto konieczne jest oddzielenie obrzeży z perforacją boczną, służącą do transportu papieru. Istnieją dwie metody dzielenia wydruków: cięcie i rozrywanie.

Stosowanie cięcia jest bardziej elastyczne, pozwala bowiem nie tylko dzielić taśmę wzdłuż, ale również tworzyć zestawy dokumentów o różnej liczbie stron, klasyfikować je według określonych kryteriów, wycinać dokumenty o różnej wysokości lub wybiórczo wycinać określone pasma. Żądaną wysokość dokumentu ustala się przed rozpoczęciem pracy za pomocą przełączników cyfrowych.

Rozrywanie jest natomiast zależne wyłącznie od istniejącej w papierze ciągłym perforacji poziomej.

Rozłączanie

Dla wydruków wykonywanych w kilku kopiach stosuje się urządzenia do rozłączania poszczególnych egzemplarzy. W przypadku stosowania kalki jest ona usuwana spomiędzy warstw papieru. Kalkę tę nawija się automatycznie na odpowiednie bębny równocześnie z odkładaniem zadrukowanych warstw papieru do oddzielnych kaset odbiorczych. Rozłączanie może być również stosowane w przypadku papieru bezkalkowego.

Kopertowanie

Kopertowanie (umieszczanie dokumentów w kopertach) może być również wykonywane automatycznie. Maszyna do kopertowania składa się z kilku stanowisk. Jedno z nich zawiera koperty, natomiast na pozostałych stanowiskach zmagazynowane są dokumenty przeznaczone do umieszczenia w kopertach. W każdym cyklu kopertowania można pobierać dokumenty z różnych stanowisk, co umożliwia umieszczenie kilku różnych dokumentów w jednej kopercie adresowanej do jednego odbiorcy.

Podstawowe maszyny do dzielenia, rozłączania i kopertowania mogą działać niezależnie albo też w systemie wyposażonym w elektroniczne sterowanie i kontrolę nad całym zestawem maszyn, obejmującym również urządzenia uzupełniające, np. do załamywania papieru, porządkowania kolejności dokumentów, zszywania (np. książeczki opłat), składania lub optycznego odczytu znaków.

Powyższe bardzo skrótowe informacje oczywiście tylko sygnalizują problem, nie wnikając w szczegóły organizacyjne i techniczne. Ostatnio pod naciskiem potrzeb coraz większych ośrodków obliczeniowych oraz rosnących rozmiarów przetwarzania można nawet zaobserwować pewne zainteresowanie tymi sprawami. Problem nie był jednak dotychczas omawiany w literaturze fachowej i tym bardziej celowe byłoby podjęcie go na łamach INFORMATYKI. Jest to bowiem jeden z bardzo istotnych i prostych sposobów zwiększających efektywność wielu systemów przetwarzania danych.

(mb)

Zapraszamy na nasze łamy

System GEORGE 2 — doświadczenia eksploatacyjne

Pierwsze próby wykorzystania Systemu GEORGE 2 podjęto w CIGM na początku 1975 r., wkrótce po ostatecznym uruchomieniu zestawu maszyny ODRA 1305.

Zaczęto wówczas stosować taśmową wersję systemu. Na skutek dość dużej początkowo awaryjności pracy jednostek pamięci taśmowej odstąpiono jednak od wdrożenia tej wersji. W połowie 1975 r. zainstalowano dwie jednostki dyskowe po 8 M znaków, co umożliwiło uruchomienie wersji bezpośredniej *(on-line)* i strumieniowej *(streamline)* tego systemu. W tym samym czasie zakończono wdrażanie pierwszego systemu transmisji dla Zjednoczenia Portów Morskich.

WYBÓR SYSTEMU OPERACYJNEGO

Wybór wariantu systemu operacyjnego wynikał z zadań ośrodka obliczeniowego. Centrum Informatyki oferuje przedsiębiorstwom gospodarki morskiej systemy przetwarzania, oparte na konfiguracji dopasowanej w miarę możliwości do potrzeb użytkowników. Przykładowo, w części przedsiębiorstw instalowane są zestawy minikomputerów wtedy, gdy ich własności techniczne i programowe zapewniają optymalną realizację funkcji systemu. W dużej części systemów wymagane jest wprowadzanie danych i otrzymywanie wyników na bieżąco lecz liczba danych wymagających bezpośredniego dostępu i zakres ich przetwarzania przekraczają techniczno-programowe możliwości minikomputerów. W tych przypadkach instalowane są monitory ekranowe przyłączane — bezpośrednio lub przez minikomputery — do zestawu komputera ODRA 1305 w Gdyni. Stąd też główny wysiłek w pracach projektowo-programowych został skoncentrowany na rozwoju tego typu systemów.

W konsekwencji eksploatacja systemów działających w trybie *on-line* jest podstawowym zadaniem ośrodka obliczeniowego w Gdyni, natomiast przetwarzanie wsadowe wykorzystuje wolną moc obliczeniową zestawu.

Głównym czynnikiem, który zadecydował jednak o odstąpieniu od wdrażania systemu GEORGE 3 było zbyt duże obciążenie pamięci operacyjnej jednostki centralnej. Przeciętny system *on-line* zajmuje 28 K słów pamięci. Jednoczesna eksploatacja trzech systemów powodowała wobec tego zajętość ponad 80 K słów pamięci operacyjnej. Wolna część tej pamięci mogła być wykorzystana więc do realizacji prac wsadowych pod nadzorem systemu GEORGE 2, a nie GEORGE 3. Rozbudowa pamięci operacyjnej powyżej 128 K słów w celu umożliwienia zainstalowania systemu GEORGE 3 nie była opłacalna ze względu na to, że na trzeciej i częściowo na drugiej zmianie ośrodek obliczeniowy dysponował rezerwami wynikającymi z mniejszego obciążenia systemami *on-line*.

Własność MOP — elementu GEORGE 3 — mogła by być wykorzystana tylko na użytek wewnętrzny Centrum, klienci zewnętrzni potrzebowali bowiem bezpośredniego dostępu do danych, a nie do systemu. System GEORGE 3 nie oferował zaś żadnych ułatwień w zakresie eksploatacji sytemów transakcyjnych typu *on-line*.

Za przyjęciem systemu GEORGE 2 przemawiały ponadto następujące czynniki:

- prostota systemu i związana z nią łatwość przeszkolenia służb eksploatacyjnych
- łatwość stopniowego włączania systemów użytkowych
- przewidywana instalacja drugiej jednostki centralnej, stanowiącej zabezpieczenie działania systemów *on-line* na wypadek awarii ODRY i możliwość przejęcia przez nią większej części przetwarzania wsadowego
- krótszy czas odpowiedzi w systemach *on-line* przy obsłudze procesora komunikacyjnego przez egzekutor w porównaniu z obsługą przez system GEORGE 3.

Wszystkie opisane powyżej czynniki złożyły się na decyzję oparcia całości przetwarzania na systemie GEORGE 2.

WDRAŻANIE SYSTEMU GEORGE 2

Wdrażanie systemu GEORGE 2 rozpoczęto od posługiwania się nim podczas testowań programów. Projektanci i programiści zostali przeszkoleni w zakresie języka opisu zadań. Opracowanych zostało szereg makrorozkazów ułatwiających opisanie typowych przebiegów, zwłaszcza kompilacji. Przygotowanie działających już systemów do pracy pod nadzorem GEORGE 2 powierzono pięcioosobowemu zespołowi w dziale technologii przetwarzania. W ośrodku obliczeniowym eksploatowanych jest około 50 systemów użytkowych, z czego 25 to systemy przedsiębiorstw resortu Handlu Zagranicznego i Gospodarki Morskiej. Konieczne było sporządzenie opisów zadań dla około 160 jednostek technologicznych, w skład których wchodziło prawie 1000 programów. Dokumentacja eksploatacyjna, szczególnie zaś dokumentacja wykonana 5—9 lat temu, zawierała niekompletne opisy programów lub nie zawierała tych opisów wcale. Podstawą do opracowania opisów zadań był wydruk z pulpitu operatora. Przebiegi systemów pod nadzorem systemu GEORGE 2 testowano na rzeczywistych zbiorach, a zadania wykonane poprawnie wykorzystywano eksploatacyjnie. Cennym ułatwieniem była możliwość prowadzenia równolegle przetwarzania bez systemu GEORGE. Błąd lub niekompletność opisu zadania mogła być korygowana przy następnym cyklu przetwarzania, nie powodując większych zakłóceń eksploatacji.

Opracowywanie opisów zadań rozpoczęto w marcu 1976 r. W lipcu zakończono wdrażanie jednostek technologicznych wchodzących w skład miesięcznych cykli przetwarzania 25 systemów. Od sierpnia do końca 1976 r. dwie osoby wdrażały pod nadzorem GEORGE 2 jednostki technologiczne związane z kwartalnymi, półrocznymi i rocznymi cyklami przetwarzania. Przygotowując stare systemy, dokonano ich częściowego unowocześnienia, zastępując sortowania taśmowe sortowaniem na dyskach. W listopadzie 1976 r. wszystkie systemy resortowe były już przetwarzane pod nadzorem systemu GEORGE 2. Dokumentację eksploatacyjną uzupełniono kartami informacyjnymi przebiegów, wydrukami opisu zadania i zbioru monitorowania.

Rozpoczęto również przetwarzanie zdalne, wykorzystując stacje ICL 7020, zainstalowane w ośrodku projektowania Centrum (odległym około 1,5 km od ośrodka obliczeniowego) i w Instytucie Budownictwa Wodnego PAN w Gdańsku. Zadania przesłane z tych stacji przetwarzano pod nadzorem GEORGE 2.

Opracowane zostały także programy rozliczające czas komputera na podstawie danych zapisywanych przez GEORGE 2 do odpowiedniego zbioru dyskowego.

W 1976 r. zestaw urządzeń w ośrodku obliczeniowym został powiększony o jednostkę centralną ICL 1904E z pamięcią o pojemności 96 K słów, trzy jednostki pamięci taśmowej PT-3, drukarkę i czytnik/perforator.

Uruchomiono także zestaw czterech jednostek dyskowych po 60 mln znaków. Stacja ICL 7020, zainstalowana pierwotnie w ośrodku projektowania w Gdyni, została przeniesiona do ośrodka regionalnego CIGM w Szczecinie.

DOŚWIADCZENIA EKSPLOATACYJNE

Wszystkie testowania nowych programów odbywają się obecnie pod nadzorem systemu GEORGE 2. Testowanie wymaga normalnie wielu czynności operatorskich, a zastosowanie systemu GEORGE 2 pozwoliło na ograniczenie tych czynności do obsługi czytnika kart, zakładania taśm, wymiany pakietów dyskowych itp., co znacznie skróciło czas przebiegu testowania.

Większość systemów użytkowych, w tym wszystkie systemy resortowe, są przetwarzane pod nadzorem GEORGE 2. Wdrażany jest pierwszy system, w którym część przetwarzania wykonywana jest u użytkownika na minikomputerze DATAPOINT 2200, stosującym protokół stacji ICL 7020, a część w ośrodku obliczenio-

wym w Gdyni. Programy GEORGE 2 (≠XKEE i ≠XKVB) stanowią integralną część tego systemu. Stare systemy wsadowe, do których w okresie wdrażania dorabiano opisy zadań, pracują poprawnie. Pojawiające się jednak czasami nieprzewidziane zatrzymania programów powodują potrzebę korekty opisów zadań.

Najefektywniejsza w użyciu jest strumieniowa wersja GEORGE 2. Pozwala ona na najlepsze wykorzystanie urządzeń wolnych — czytników i drukarek, które normalnie są wąskim gardłem ograniczającym przepustowość zestawu przy pracy wsadowej.

Operacje wczytywania danych, przetwarzania i wydruku są rozdzielone i wzajemnie niezależne. Należy zwrócić jednak uwagę na to, że system GEORGE 2 jest systemem jednoprogramowym. Zadania wykonywane są zgodnie z ich kolejnością we wsadzie. Nie ma możliwości przyspieszenia wykonania określonego zadania, jeśli poprzednie nie zostały zakończone. Jeden wsad zajmuje jeden lub więcej zbiorów pośrednich, moduły centralne otwierają zbiory pośrednie w sposób przypadkowy i w związku z tym zdarza się, że wykonywane są najpierw zadania wchodzące w skład wsadu wczytanego później od pozostałych. Uruchomienie tylu modułów centralnych, ile jest wczytanych wsadów, jest nie zawsze możliwe. Dopiero ostatnio wydane przez ICL wersje programów ≠XKEE i ≠XKVB umożliwiają wyróżnienie wsadów przesłanych z urządzeń zdalnych. Przedtem zdarzało się, że praca przesłana ze stacji ICL 7020 musiała czekać na wykonanie zadań lokalnych.

Działanie równoległe pracujących modułów centralnych nie jest wzajemnie koordynowane i są one obsługiwane przez egzekutor podobnie jak inny program. W związku z tym może zdarzyć się, że np. dwa moduły centralne zażądają jednocześnie zwiększenia obszaru pamięci. Usunięcie jednego z nich spowoduje konieczność powtórzenia aktualnie wykonywanego przezeń zadania. Podobna sytuacja może wystąpić w przypadku niedostatecznej liczby jednostek pamięci taśmowej lub dyskowej.

Właściwe przygotowanie prac pozwala na uniknięcie opisanych wyżej trudności. Znacznie wzrasta więc rola zespołu przygotowującego obliczenia. Musi on odpowiednio pogrupować zadania, uwzględniając wielkość koniecznej dla pamięci programów, liczbę urządzeń pamięci masowej i czas ich wykorzystywania. Kierownik zmiany powinien na bieżąco decydować o konfiguracji programów GEORGE 2, podziale prac do wykonania pod nadzorem wersji *on-line* oraz wersji strumieniowej itp. Niespełnienie tych wymogów obniża efektywność wykorzystania komputera.

Podobnie konieczne jest uzgodnienie charakteru prac przesyłanych ze stacji ICL 7020 i odpowiednie powiązanie terminów wczytywania i wykonywania zadań lokalnych z okresami seansów łączności. Przesyłanie zadań przypadkowo, może spowodować tak znaczne opóźnienie ich wykonania, że użytkownik będzie mógł otrzymać wydruki dopiero podczas następnego seansu łączności.

W trakcie wdrażania systemu GEORGE 2 dużą zaletą okazała się jego elastyczność, umożliwiająca jednoczesne przetwarzanie pod nadzorem GEORGE 2 i pod bezpośrednią kontrolą egzekutora. Ośrodek obliczeniowy podejmował się bowiem wykonywania prac dla klientów spza resortu i nie przygotowanych do przetwarzania pod nadzorem GEORGE 2.

Mając na uwadze ograniczenia systemu GEORGE 2, ostatnio przystąpiono do wdrażania systemu GEORGE 3 w nowo powstałym ośrodku w Gdańsku.

Zamierza się uzyskać doświadczenia z eksploatacji mieszanki różnego rodzaju systemów (partiowe, lokalne, partiowe zdalne, transakcyjne) dla przeprowadzenia analizy porównawczej z konfiguracją ośrodka w Gdyni, pracującego pod nadzorem systemu GEORGE 2. Wyniki tej analizy powinny pozwolić na określenie warunków, w jakich korzystniej jest prowadzić pracę zestawu obliczeniowego pod nadzorem GEORGE 2 lub GEORGE 3.

Piotr BRZEZICKI, Roman STYGAR
Centrum Informatyki Gospodarki Morskiej
Gdynia

Bibliografia wydawnictw polskich z dziedziny informatyki

● **Problemy informatyki prawniczej — MRÓZEK A. Wyd. Uniwersytetu Mikołaja Kopernika, Toruń 1977 r., s. 206, cena 11 zł**

Teoretyczne podstawy informatyki prawniczej — pojęcia podstawowe. Informacyjna struktura procesu działania prawa. Zagadnienia techniczne automatycznego przetwarzania danych. Budowa banków danych, ze szczególnym uwzględnieniem potrzeb dokumentacji prawniczej. Problemy funkcjonowania prawniczych banków danych. Zautomatyzowane banki danych zawierających informacje o faktach wielonormatywnych. Automatyzacja procesów podejmowania decyzji o charakterze konkretnoindywidualnym. Organizacyjne i prawne problemy automatyzacji w zarządzaniu. Kierunki rozwoju automatycznego przetwarzania danych w procesach zarządzania w PRL. Przypisy. Słownik podstawowych pojęć stosowanych w pracy.

Skrypt przeznaczony jest dla studentów wydziałów prawa i administracji, którzy począwszy od roku akademickiego 1976/77 po raz pierwszy będą mieli okazję uczestniczenia w wykładach z zakresu tzw. informatyki prawniczej.

● **Zasady budowy systemów informowania kierownictwa — studium teoretyczne — FLAKIEWICZ W. Wyd. Ośrodka Badawczo-Rozwojowego Informatyki, Warszawa 1977 r., s. 157**

Zarys metody budowy SIK (systemy informowania kierownictwa): problem zdefiniowania zakresu pojęcia SIK, SIK a organizacja banku danych, niektóre bieżąco obserwowane problemy budowy SIK I, zaady tworzenia SIK II, obszar prac teoretyczno-metodologicznych związanych z budową SIK II.

Koncepcja języka L oraz opis systemu podejmowania decyzji: założenia wyjściowe, ogólna koncepcja języka stosowanego do opisu systemu, konstrukcja opisu systemu obserwowanego, ogólne cechy języka L, system obserwator i analiza systemu w języku L, opis decyzji w języku L.

Materiały przeznaczone są dla kadry kierowniczej instytucji, pracowników zajmujących się teorią i praktyką zarządzania oraz projektantów systemów EPD.

● **Od teorii mnogości do algebry logiki. Zastosowanie algebry logiki w technice cyfrowej — SIEGFRIED W. J. Tłum. wyd. niem. z 1975 r. WKiŁ, Warszawa 1977 r., s. 131, cena 17 zł**

Teoria mnogości: liczby i zbiory, działania na zbiorach, binarna technika cyfrowa. Algebra logiki: podstawy z reguły algebry logiki, zapis funkcji przełączających. Funkcje przełączające: optymalna forma funkcji przełączającej, analiza funkcji przełączających. Metoda graficzna — siatki Karnaugha: siatki Karnaugha i wykresy Venna, budowa siatki Karnaugha dla trzech zmiennych, siatki Karnaugha dla czterech, pięciu, sześciu zmiennych, analiza i synteza funkcji przełączających przy pomocy siatek Karnaugha, kombinacje nadmiarowe (zbyteczne).

Dodatek: zestawienie najważniejszych reguł teorii mnogości i algebry logiki, symbole matematyczne algebry logiki, symbole schematowe elementów cyfrowych. Każdy rozdział zawiera ćwiczenia i zadania sprawdzające. Na końcu książki podano ich rozwiązanie. W książce podano w przystępny sposób podstawy techniki cyfrowej.

Książka przeznaczona jest dla osób o zainteresowaniach technicznych. Może być pomocna dla uczniów średnich szkół technicznych oraz studentów.

● **Elektroniczna technika obliczeniowa i język programowania FORTRAN IV — BOSSAK M., GROCHMAL S. Wyd. Politechniki Rzeszowskiej im. Ignacego Łukasiewicza, Rzeszów 1977 r., s. 165, cena 16 zł**

Elektroniczna technika obliczeniowa: elektroniczne maszyny analogowe, elektroniczne maszyny cyfrowe. FORTRAN IV: stałe, zmienne i tablice, funkcje standardowe, wyrażenia, instrukcje, podprogramy (procedury), przygotowanie programu do uruchomienia, przykładowe programy i procedury. Słownik wybranych terminów z zakresu ETO.

Skrypt przeznaczony jest dla studentów Instytutu Lotnictwa i Instytutu Budowy Maszyn do zajęć wykładowych i laboratoryjnych z przedmiotów „elektroniczna technika obliczeniowa„ i „informatyka".

Problemy układania harmonogramów

Chciałbym tymi uwagami rozwiać pewne wątpliwości, jakie nasunąć się mogą przy czytaniu publikacji L. Świderskiego *), a przy okazji zasygnalizować, gdzie w analizie sieciowej tkwią jeszcze rzeczywiste problemy. Uwagi te są o tyle aktualne, że w druku znajduje się polskie tłumaczenie książki „Teoria zaszeregowania zadań", pod redakcją E. G. Coffmana, Jr., która omawia aktualne problemy badawcze w deterministycznej teorii szeregowania (tj. w teorii układania harmonogramów).

W pracy L. Świderskiego można wyróżnić dwie zasadnicze części. W pierwszej przedstawione są problemy, metody rozwiązywania i przykłady obliczeń (44 strony), a druga jest zbiorem programów w języku GIER ALGOL-4, realizujących metody przedstawione w pierwszej części (51 stron).

● Omówione zostały sposoby układania harmonogramów dla produkcji jednowyrobowej, wielowyrobowej i seryjnej. Te trzy przypadki w obu częściach opracowania zostały potraktowane oddzielnie, chociaż z matematycznego punktu widzenia (a tym bardziej dla maszyny cyfrowej) dopuszczalne są przecież serie złożone z jednego elementu i produkcje wielowyrobowe polegające na wykonaniu jednego wyrobu. Podobnie, w sposób raczej sztuczny, powszechnie stosowane pojęcie harmonogramu (np. w metodzie drogi krytycznej) rozbite jest w tej pracy na trzy rodzaje harmonogramów.

● Przedstawione metody obliczeniowe są realizacjami tych samych zależności, z których korzysta metoda drogi krytycznej (CPM), znana, jak sam Autor pisze, od 1956 roku. Sieciowy model przedsięwzięcia jest przedstawiony w postaci drzewa skierowanego, które oczywiście jest szczególnym przypadkiem bezkonturowego grafu skierowanego.

Proponowany sposób wyznaczania harmonogramów uwzględnia między innymi możliwość występowania w wyrobie jednakowych podzespołów. Łatwo można zauważyć, w jaki sposób drzewo skierowane z jednakowymi poddrzewami może być zastąpione równoważnym, w sensie żądanych wyników, modelem sieciowym w postaci bezkonturowego drzewa skierowanego. Podobnie można sobie poradzić z modelami dla produkcji wielowyrobowej i seryjnej.

Tak więc rozpatrywane modele są szczególnymi przypadkami modeli rozwiązywanych za pomocą CPM.

● Całkowicie błędne jest przekonanie Autora, że zarówno w opisach metod sieciowych, jak i w samych maszynowych realizacjach używane są jedynie macierzowe sposoby opisu grafu. Aby się o tym przekonać wystarczy sięgnąć do niektórych z cytowanych opracowań, np. w książce A. Z. Idźkiewicza „PERT. Metody analizy sieciowej" nie występuje żadna macierz. Podobna sytuacja jest wśród metod obliczeniowych, patrz np. procedury 34 i 35 w książce J. Kucharczyka i M. M. Sysło — „Algorytmy optymalizacji w języku ALGOL 60", PWN, Warszawa 1975.

Wydaje mi się, że nawet w programach standardowych komputera GIER, z którymi Autor porównywał swoje programy, używane są także niemacierzowe sposoby zapisu grafu.

Autor proponuje dwa sposoby zapisu struktury drzewa, z których pierwszy oparty jest na znanym kodowaniu drzew za pomocą beznawiasowej symboliki Łukasiewicza, a drugi jest szczególnym przypadkiem powszechnie stosowanej listowej reprezentacji grafu.

We wszystkich opisach algorytmów i w programach zastosowany jest ten drugi sposób, jako wygodniejszy. Przy czym nigdzie wyraźnie nie jest powiedziane (i nie badane), że listy powinny być tak uporządkowane, aby obliczenia mogły przebiegać poprawnie „od dołu do góry". Nie wystarcza wyjaśnienie, zwłaszcza dla wyrobów o bardzo wielu częściach, że „lista części może być układana bądź *piętrami*, bądź też *gałęziami* schematu wyrobu".

● Jednym z podstawowych celów tej pracy było zbadanie, w jakim stopniu zastosowanie innej metody opisu modelu sieciowego ma wpływ na efektywność układania harmonogramów za pomocą komputera. Porównano w tym celu programy standardowe komputera GIER 4 z programami zaproponowanymi przez Autora. Okazało się, że te drugie są od 4 do 10 razy szybsze, ale jakie to może mieć znaczenie, jeżeli programy standardowe zostały potraktowane jako czarne skrzynki, o których wiadomo jedynie, że dają takie a nie inne wyniki dla odpowiednio przygotowanych danych? Obecnie, kiedy problematyka złożoności obliczeniowej jest tak popularna i szeroko rozwijana, takie podejście do badania efektywności algorytmów jest niewystarczające.

Przytoczone porównania czasów obliczeń mogą świadczyć jedynie na korzyść programisty. Po pobieżnym jednak przejrzeniu algorytmów (nie znam bowiem GIER ALGOLU 4) wydaje mi się, że można jeszcze wiele zrobić dla ich przyspieszenia. Np. w każdym programie oblicza się wielkości $3i-2$, $3i-1$, $3i$, które przecież dla kolejnych wartości $i = 1, 2, \ldots, n$ są kolejnymi liczbami od 1 do $3n$, a więc mogą być wyznaczane bez użycia mnożeń.

Jeszcze gorzej wygląda zasadniczy fragment obliczeń w pierwszym programie, gdzie liczba operacji (np. porównań) jest $O(n^2)$, a bez trudu może być zmniejszona do $O(n)$.

● Jednym z celów, jakie stawia sobie Autor na przyszłość, jest zainteresowanie osiągniętymi wynikami producenta komputerów, zakładów ELWRO. Samo przeprogramowanie istniejących programów, jak pisze Autor, z języka GIER ALGOL 4 nie powinno nastręczać większych kłopotów. Po pierwsze jednak, w przypadku dwóch programów, napisanych dla pamięci bębnowych i karuzelowych, jest to bezcelowe dla komputerów produkowanych przez ELWRO, a w pozostałych przypadkach jest to rzeczywiście bardzo proste. Np. w pierwszym programie, którego wydruk zajmuje 4 strony, należałoby w tym celu wziąć pod uwagę jedynie instrukcje zajmujące mniej niż pół strony, gdyż wszystkie inne mają na celu jedynie przygotowanie danych i wyników i są charakterystyczne tylko dla komputerów GIER (w kraju jest tylko jeden egzemplarz tej maszyny).

Podajemy dla informacji, że wszystkie aktualnie produkowane przez ELWRO komputery wyposażone są w programy dla analizy sieci.

● W pracy nie wspomina się o związkach między sieciami czynności a sieciami stanów, chociaż skądinąd wiadomo, że wybór któregoś z tych sposobów reprezentacji może mieć istotny wpływ na sposób konstrukcji sieci i później na samą jej analizę.

Innym problemem narzucającym się w sposób bardzo naturalny przy rozpatrywaniu wyrobów z jednakowymi podzespołami jest gospodarka zasobami. Bierze się to stąd, że zwykle te same części czy też podzespoły wytwarzane są na tych samych maszynach przy wykorzystaniu tych samych zasobów.

Reasumując, w świetle ogólnie znanych i dostępnych metod omawiane oprogramowanie nie jest odzwierciedleniem obecnego stanu analizy sieci i teorii szeregowania. Dlatego nie polecałbym tej pracy tym, którzy chcą się zapoznać z podstawami istniejących metod sieciowych i metod układania harmonogramów.

Maciej M. SYSŁO

*) L. Świderski: „Niektóre zagadnienia związane z układaniem harmonogramów za pomocą maszyn cyfrowych". PWN, Warszawa 1977, 95 stron, nakład 1200 egz.

Jak ZSAP wspomagają prace projektowe

Stosowanie komputerów w pracach projektowych ma już sporą tradycję. Początkowo pisano stosunkowo małe, pojedyncze programy, przeznaczone do wykonywania najbardziej żmudnych fragmentów obliczeń, które stopniowo obejmowały coraz większe obszary prac projektowych. Następnie tworzono pakiety programowe, składające się ciągle jeszcze z formalnie samodzielnych programów. Z czasem jednak zaczęto nadawać im postać mniej lub bardziej kompleksowych systemów.

Wraz z rozwojem sprzętu komputerowego powstawały coraz bardziej rozbudowane systemy projektujące. W założeniu swoim systemy te miały odciążyć użytkownika-projektanta od żmudnych i pracochłonnych czynności, już nie tylko obliczeniowych (numerycznych), ale i od prac edytorskich, graficznych, wyszukiwania i kontroli danych itp. Okazało się jednak, że posługiwanie się tak dużymi i stosunkowo sztywno zbudowanymi systemami jest dla projektantów, również uciążliwe, a ponadto powoduje z ich strony różnego rodzaju opory przed narzucaniem gotowych rozwiązań i eliminowaniem twórczej inicjatywy, a więc przed nadmiernym zautomatyzowaniem czynności koncepcyjnych.

Teoretyczne i praktyczne poszukiwania wyjścia z tego impasu doprowadziły do sformułowania m. in. dwu pozornie sprzecznych, a jak się właściwie okazuje, uzupełniających się koncepcji: idei języków problemowo zorientowanych (JPZ), co usuwa barierę niezrozumienia przez projektanta form pracy z komputerem oraz idei projektowania wspomaganego komputerem — jako przeciwstawnej rodzącemu się zagrożeniu automatyzacji.

Zasygnalizowane problemy w zasadzie poruszane są dotychczas bądź na specjalistycznych konferencjach i seminariach, bądź w publikacjach o charakterze artykułowym zamieszczanych również w specjalistycznych czasopismach. Brak jest natomiast wyczerpujących publikacji książkowych, które mogłyby służyć każdemu czytelnikowi zainteresowanemu problematyką metodyki projektowania, komputerowo wspomaganego, bez względu na branżowy charakter tegoż projektowania. Dlatego z dużym zainteresowaniem powinny się spotkać dwie prace Marii Wolpe [1], [2] omawiające Zintegrowane Systemy Automatyzacji Projektowania pod kątem tzw. użytkownika czynnego (programisty podsystemowego).

Pierwsza z tych pozycji [1] jest rodzajem monografii. Autorka dużo wagi poświęca zobrazowaniu zarówno samej idei ZASP, jak i historii jej powstawania, przy czym robi to w kontekście porównania trzech wymienionych w tytule systemów: amerykańskiego ICES, czeskiego SAPRO i brytyjskiego GENESYS. Natomiast pozycja druga [2] jest podręcznikiem przeznaczonym dla użytkowników pracujących w SAPRO. W książce tej przedstawiony jest system SAPRO zainstalowany w Polsce na komputerze ICL System 4-50/70 w ZIPO Gdańsk. Pozwala to wniknąć w zasady pisania nowych podsystemów i uchwycić istotę problemów, występujących przy takiej pracy. Jest to pozycja o zasięgu szerszym niż wynikałoby to z zamierzeń autorki.

W obu pozycjach sporo miejsca poświęca autorka na omówienie zasadniczych aspektów budowy i działania ZSAP: rozdział 1 w [1] oraz rozdziały 1 i 7 w [2].

Strukturę ZSAP można szkicowo charakteryzować jako trzypoziomową:

- poziom najniższy — systemowy — zarządzający pracą całego systemu oraz zawierający środki programowe do pisania (tworzenia) poszczególnych, problemowo zorientowanych podsystemów.
  Są to:
  - — rozszerzony język algorytmiczny (rozszerzony FORTRAN)
  - — język opisu podsystemów i rozkazów Języków Problemowo Zorientowanych (generator translatorów JPZ)
  - — język opisu danych globalnych
- poziom pośredni — podsystemowy — zawierający środki do realizacji konkretnych programów obliczeniowych z danej dziedziny:
  - — skompilowane moduły programowe realizujące konkretne zadania
  - — JPZ danego podsystemu
- poziom najwyższy — użytkowy — programy zazwyczaj jednorazowego zastosowania napisane przez projektanta, tzn. użytkownika biernego w danym JPZ.

Analizując działanie ZSAP od strony inżyniera projektującego, korzystającego z wyspecjalizowanego JPZ — można sądzić, że mamy do czynienia z daleko posuniętą automatyzacją prac projektowych. Jeżeli jednak rozpatrzy się ten sam ZSAP od strony całego aparatu programistycznego, umożliwiającego pisanie nowych podsystemów, to trzeba stwierdzić, że prezentowane środki programowe spełniają postulaty projektowania wspomaganego komputerem.

Specjalne rozwiązania na poziomie systemowym ZSAP zapewniają realizację dwóch podstawowych cech, dzięki którym można mówić o systemie, jako wewnętrznie zintegrowanym:

- istnieje możliwość przenoszenia danych nie tylko w ramach jednego programu, co zapewnia każdy język algorytmiczny, ale i pomiędzy programami wchodzącymi w skład jednego podsystemu, a także pomiędzy programami wchodzącymi w skład różnych podsystemów
- istnieje możliwość wywoływania w jednym programie podprogramów wchodzących w skład innego skompilowanego programu.

Ponadto, na szczególną uwagę zasługuje „prawdziwa" — zdaniem autorki — otwartość systemu. Otwartości tej właśnie służy cały aparat tworzenia podsystemów, owe wspomniane trzy języki poziomu systemowego. Dlatego też opisowi tych języków poświęca autorka kolejne rozdziały swoich książek.

Rozszerzony język programowania omawiany jest w rozdziałach drugich obu publikacji. Przy czym zgodnie z różnym ich charakterem, w [1] scharakteryzowano wszystkie trzy języki: SAPROTRAN, ICETRAN i GENTRAN, występujące między nimi różnice oraz ich rozszerzenia w stosunku do „czystego" FORTRANU (który jest bazą wszystkich tych języków), takie jak:

- — tablice dynamiczne
- — programy dynamiczne i statyczne
- — rozszerzone instrukcje podstawiania.

Rozszerzone języki programowania służą do pisania programów realizujących poszczególne zadania obliczeniowe i są narzędziem pracy użytkownika czynnego. Program napisany w rozszerzonym języku zostaje przez prekompilator właściwego ZSAP sprowadzony do postaci „czystego" FORTRANU. A w tej postaci zostaje przekazany standardowemu kompilatorowi FORTRANU danego komputera.

W publikacji [2] — problematyka rozszerzonego języka programowania — SAPROTRANU, potraktowana jest bardziej wnikliwie. Omówione są wszystkie instrukcje różniące SAPROTRAN od FORTRANU, ich formaty, zasady

---

[1] Maria Wolpe: Zintegrowane Systemy automatyzacji projektowania. SAPRO, ICES, GENESYS. Arkady. Warszawa 1977, str. 189, 20 rys., 24 tabl., 21 poz. lit., nakład 3180

[2] Maria Wolpe: System automatyzacji projektowania SAPRO. Podręcznik programowania podsystemowego. Prace PW. Wydz. Inż. Lądowej, Instytut Dróg i Mostów, Warszawa 1977, str. 261, 3 załączniki, 10 rys., tab. 20, poz. lit. 8, nakład 330

działania i przykłady zastosowań. Ma to szczególne znaczenie dla praktycznego korzystania z podręcznika. Uzupełnieniem tego rozdziału są rozdziały 6 i 7 pracy [2], omawiające ogólne zasady pracy w systemie SAPRO oraz tzw. rozkazy systemowe SAPRO, jak również rozdział 8 — podający opis błędów. Dopiero łączne informacje z tych czterech rozdziałów (rozdz.: 2, 6, 7, 8) pozwalają na samodzielną pracę z komputerem przy pisaniu programów (modułów programowych) w SAPROTRANIE.

Opracowywanie nowych (lub modyfikacja istniejących) podsystemów, oprócz napisania odpowiednich modułów programowych realizujących poszczególne zadania wymaga opisu organizacji podsystemu oraz opisu jego JPZ, jak również opisu struktury i pracy danych globalnych tego podsystemu oraz sposobów ich włączania w ogólnosystemowe struktury danych. Każdy z ZSAP, a więc i SAPRO dysponuje do tych celów specjalnymi środkami programowymi. W SAPRO są to specjalne języki: DEFPO do opisu podsystemów i rozkazów JPZ oraz DEFDAT do opisu danych. W istocie rzeczy są to również JPZ, ale przeznaczone tylko dla użytkowników czynnych, a więc niedostępne dla użytkowników biernych. Charakterystyka i opis zasad działania tych języków zawarte są w rozdziałach 4 i 5 pracy [2]. Podano tam wszystkie instrukcje, ich formaty ogólne i możliwe formy szczegółowe. Dobre operowanie przykładami czynią materiał zawarty w tych rozdziałach szczególnie przydatnym. W pracy [1] omówiono jedynie języki opisu podsystemów (rozdział 4) odpowiednio: DEFPO w SAPRO, CDL w ICES oraz instrukcje opisu podsystemów i generowania rozkazów JPZ występujące w GENTRANIE, gdyż w systemie GENESYS ten aparat programistyczny nie został wydzielony w osobny język. Informacje na temat języków opisu podsystemów zamieszczone w pracy [1] są mniej dokładne od analogicznych podanych w [2]. Szczególnie silnie odczuwa się brak większej ilości przykładów, które ustępują miejsca porównaniom pomiędzy opisywanymi systemami.

Poważnym mankamentem w pracy [1] wydaje się również całkowite pominięcie języka opisu danych. Utrudnia to uchwycenie paru problemów istotnych dla samej idei ZSAP, takich jak: przenoszenie danych pomiędzy podsystemami, praca ze zbiorami typu SAPGEN i podobne.

Sporym mankamentem obu opracowań jest również zbytnie ograniczenie informacji o JPZ i całkowite zrezygnowanie z materiału przykładowego w tym zakresie. Wprawdzie w obu książkach są rozdziały (rozdziały 3) zatytułowane: „Języki problemowo zorientowane", ale zawierają one jedynie zasady ogólne. Wskutek tego czytelnik nie posiadający wiadomości na temat JPZ nie jest w stanie wyrobić sobie głębszego poglądu na ich istotę, ani na istotę związku całego aparatu systemowego ZSAP i aparatu poszczególnych podsystemów z najwyższą — właściwą warstwą użytkową. Rzecz jasna, zamierzeniem autorki było takie właśnie ustawienie problemów. Zakłada ona, iż czytelnik jej książek orientuje się w zasadach pracy w JPZ, np., że wykorzystywał on już takie języki, natomiast omawiane [1] i [2] publikacje są adresowane do twórców nowych podsystemów — użytkowników czynnych. Są to tej klasy inżynierowie, że ich pracę z ZSAP można zakwalifikować jako wspomaganie przez system prac projektowych. Z tego punktu widzenia są to pozycje bardzo cenne dla polskiego czytelnika. Warto również podkreślić, że dostarczają one wiadomości o środkach programowych, pozwalających na pisanie podsystemów dotyczących dowolnej dziedziny projektowania inżynierskiego, aczkolwiek omówione ZSAP powstawały na potrzeby szeroko pojmowanego projektowania budowlanego.

Obie książki dostarczają również wielu ciekawych informacji na temat pierwszej warstwy — systemowej, aczkolwiek nie są tej sprawie poświęcone. Cytowana literatura tematu (zwłaszcza w pracy [1]) może być interesująca dla wielu czytelników pragnących bliżej zapoznać się z omawianą problematyką.

S. BOŃKOWICZ-SITTAUER

Bibliografia wydawnictw polskich z dziedziny informatyki

● **Laboratorium z informatyki (zbiór zadań, algorytmów i programów) — SZYPULSKA M. — red. nauk. Wyd. Wyższej Szkoły Morskiej, Gdynia 1977 r., s. 186, cena 19 zł**

Język BASIC (wersja HP 9830 A). Elementy programowania w języku BASIC (HP 9830 A). Metody numeryczne. Wybrane przykłady zastosowań: operacje na zbiorach danych, obliczenia nawigacyjne, analiza układów elektronicznych i elektrotechnicznych, analiza układów mechanicznych, przetwarzanie danych.

Skrypt przeznaczony jest dla studentów Wyższej Szkoły Morskiej i służyć ma jako pomoc dydaktyczna w zakresie zajęć laboratoryjnych z informatyki. Celem opracowania jest przekazanie podstawowych wiadomości o formułowaniu problemów oraz samodzielnym opracowywaniu i użytkowaniu programów komputerowych w języku konwersacyjnym BASIC.

● **Elektroniczna technika obliczeniowa. Programowanie w języku ALGOL 60 — BIERKOWSKA A. Wyd. Politechniki Rzeszowskiej im. Ignacego Łukasiewicza, Rzeszów 1976 r., s. 95, cena 4 zł**

Wprowadzenie teoretyczne. Teoria algorytm. Schematy blokowe. Programowanie w języku ALGOL 60. Ogólne informacje o programowaniu w języku FORTRAN.

Skrypt przeznaczony jest dla studentów Instytutu Technologii Chemicznej.

● **Wyszukiwanie informacji w zbiorach pamięci wewnętrznej i pamięciach pomocniczych — CICHORSKI J. Wyd. Ośrodka Badawczo-Rozwojowego Informatyki, Warszawa 1977 r., s. 78**

Organizacja fizyczna zbiorów danych. Metody wyszukiwania informacji względem jednego klucza. Metody wyszukiwania informacji względem wielu kluczy.

Materiały przeznaczone są dla projektantów systemów EPD.

● **Systemy cyfrowe automatyki przemysłowej. T. 1. Sprzęt i oprogramowanie — NIEDERLIŃSKI A. WNT, Warszawa 1977, s. 356, cena 60 zł**

Sprzęt systemów cyfrowych automatyki: układ wejść analogowych, układ wejść cyfrowych, jednostka centralna, pamięć zewnętrzna, urządzenia obsługi systemu, układ wyjść analogowych, układ wyjść cyfrowych, urządzenia łączności. Oprogramowanie systemów cyfrowych automatyki: koszt oprogramowania, modułowość oprogramowania, oprogramowanie podstawowe (system operacyjny), oprogramowanie użytkowe ogólne, oprogramowanie użytkowe specjalistyczne, języki programowania. Książka zawiera podstawowe wiadomości z powyższego zakresu najbardziej reprezentatywne w okresie do końca 1974 r.

Książka jest przeznaczona dla projektantów i użytkowników systemów cyfrowych automatyki oraz dla pracowników inżynieryjno-technicznych pragnących zapoznać się z podstawowymi problemami automatyzacji kompleksowej.

● **Systemy cyfrowe automatyki przemysłowej T. 2. Zastosowania — NIEDERLIŃSKI A. WNT, Warszawa 1977, s. 428, cena 70 zł**

Wstęp: struktura systemów automatyzacyjnych, przykład systemu automatyzacyjnego, kolejność uruchamiania systemu automatyzacyjnego. Zbieranie i przetwarzanie zmiennych procesowych: zbieranie zmiennych procesowych, przetwarzanie podstawowe zmiennych procesowych, przetwarzanie specjalistyczne zmiennych procesowych ciągłych, przetwarzanie zmiennych procesowych nieciągłych. Sygnalizacja, kontrola i dokumentacja przebiegu procesu: system sygnalizacyjno-alarmowy, system kontroli, pewne aspekty dokumentacji przebiegu procesu, przykłady. Regulacja automatyczna: podstawowe struktury układów regulacji cyfrowej, model układu regulacji cyfrowej bezpośredniej, podstawowe algorytmy regulacji cyfrowej bezpośredniej, złożone algorytmy regulacji cyfrowej. Kompensacja automatyczna: podstawowe algorytmy kompensacji automatycznej, przykłady zastosowań kompensacji automatycznej, regulacja automatyczna z kompensacją. Sterowanie binarne: struktura algorytmów sterowania binarnego, uzasadnienie celowości stosowania systemów cyfrowych do sterowania binarnego, języki problemowe dla celów sterowania binarnego. Optymalizacja procesu: elementy problemu optymalizacyjnego, przykłady formułowania problemów optymalizacyjnych, programowanie liniowe, programowanie nieliniowe, metody wspinaczkowe optymalizacji, dekompozycja problemów optymalizacyjnych.

Książka jest przeznaczona dla projektantów i użytkowników systemów cyfrowych automatyki oraz dla pracowników inżynieryjno-technicznych pragnących zapoznać się z podstawowymi problemami automatyzacji kompleksowej.

Oprac. A.K.

ZYGMUNT RYZNAR
Przedsiębiorstwo Projektowo-Technologiczne Przemysłu Lotniczego i Silnikowego „PZL — Kraków"

Bazy danych i banki danych

W poprzednich pozycjach cyklu poznaliśmy cechy charakterystyczne zbiorów klasycznych i aspekty teoretyczne struktur danych. Uzasadniliśmy również celwość realizowania zastosowań w oparciu o koncepcję wspólnej bazy danych. W niniejszym artykule postaramy się dokonać klasyfikacji baz danych. W celu uniknięcia ewentualnych niejasności terminologicznych sprecyzujemy na wstępie pojęcia zbioru, bazy danych i banku danych, umawiając się równocześnie, że wobec różnorodnych (nie zawsze zgodnych ze sobą) definicji spotykanych w literaturze, podane tutaj określenia nosić będą charakter roboczy, przydatny szczególnie w kontekście niniejszego artykułu.

Przez **zbiór danych** rozumiemy fizyczną maksymalną jednostkę danych, obsługiwaną z reguły przez system operacyjny komputera [1]), który wyznacza jej obszary pamięciowe zewnętrzne (np. na dyskach) i wewnętrzne (bufory wejścia—wyjścia), dostarcza procedury organizacji, dostępu itp.

Od strony strukturalnej w zbiorze wyróżnia się następujące elementy fizyczne: znak lub bajt, pole, zapis, blok. Ich cechą charakterystyczną jest to, że składają się one z ciągu bezpośrednio położonych obok siebie (przyległych) komponentów, np. bajt składa się z ciągu bitów, pole z ciągu bajtów, zapis z ciągu pól, blok z ciągu zapisów.

Baza danych jest makymalną logiczną jednostką danych, obsługiwaną przez system zarządzania bazą danych, który zabezpiecza realizację powiązań logicznych pomiędzy elementami danych niekoniecznie znajdującymi się obok siebie (mogą się one mieścić np. w różnych zbiorach).

Opis elementów i ich powiązań (tworzących logiczne zestawy danych) w odniesieniu do bazy danych może być wydzielony poza programy aplikacyjne (na przykład do zbioru zwanego słownikiem danych). Uzyskuje się w ten sposób niezależność danych, dzięki której zmiany strukturalne danych nie muszą powodować zmian w programach użytkownika. W szerokim ujęciu opis danych może być zaliczany do bazy danych, gdyż bez niego dane są nierozpoznawalne (tak w sensie identyfikacji elementów, jak i ich powiązań). Fizycznie opis danych wchodzi do bazy w przypadku umieszczania opisu danych wśród danych (np. w bazach archiwalnych) przy tworzeniu tzw. zapisów samodefiniowalnych.

Bank danych jest użytkową jednostką eksploatacyjną, obejmującą dane znajdujące się w bazie, ich opis i uogólnione (w miarę uniwersalne) oprogramowanie zapewniające eksploatację bazy (baz) przez różne programy użytkowe. Oprogramowanie to występuje najczęściej pod nazwą „system zarządzania bazą danych". Bardziej szczegółową specyfikację funkcji tego systemu podaliśmy w poprzednim artykule cyklu.

Dla porównania powyższych definicji przytoczymy kilka innych definicji pochodzących od autorów zagranicznych, które są zapewne mało znane w środowisku naszych informatyków. Brak miejsca nie pozwala w niniejszym artykule na pełną dyskusję terminologiczną.

W. E. Bender np. pisze [6] o bazie danych następująco: „W swojej najprostszej formie, **baza danych jest zestawem danych**, dla których organizacja przedsiębiorstwa wymaga ustalenia **pewnych powiązań i ich utrzymania.** System zarządzania bazą danych **strukturyzuje** i utrzymuje te relacje" (tłumaczenia i wytłuszczenia w tekście — autora).

H. Wedekind określa [4] **bazę danych jako zestaw zbiorów** (relacji) podporządkowany pewnym ograniczeniom i strukturom wewnętrznym *intra-file structures).* Ograniczenia dotyczą semantycznej i syntaktycznej prawidłowości danych, zaś struktury odnoszą się do techniki tworzenia zbiorów pierwotnych (ang. *master)* i wtórych (ang. *derived).*

W związku z powyższą definicją nasuwa się pytanie **kiedy baza danych może składać się z więcej niż jednego zbioru.** Odpowiedź nie jest prosta. Należy bowiem przedtem określić kryteria podziału danych na szereg zbiorów. Jeśli w zbiorach utrzymywane są jednolite zasady strukturalne (układ kluczy pierwotnych, podzielność lub niepodzielność zapisów, rodzaje więzi łańcuchowych, itp.) i powstały one z przyczyn „technicznych" (np. z ograniczeń pamięci dyskowych), wówczas możemy mówić o bazie wielozbiorowej, korzystającej z tego samego schematu (słownika danych) i z tych samych procedur utrzymywania struktur.

Interesujące wskazówki terminologiczne zawiera praca J. Martina [3], który definiuje bazę danych jako **„zestaw wystąpień wielu typów zapisów** (rekordów), zawierający powiązania pomiędzy tymi zapisami, agregacjami danych i pozycjami danych", zaś **zbiór danych** jako „nazwany (ang. *named),* **zestaw wszystkich wystąpień danego typu zapisu".**

Dla porównania przytoczymy również definicję bazy danych, opracowaną przez organizację CODASYL: „Baza danych składa się ze wszystkich zapisów, zbiorów i obszarów, które są sterowane przez określony schemat" [1, s. 34]. Schemat zawiera nazwy i opisy elementów bazy danych oraz ich powiązań. Przy okazji warto zaznaczyć, że „zbiór" (ang. *set*) w ujęciu powyższym oznacza zestaw logiczny nie tylko na poziomie zbioru fizycznego lecz również na poziomie grup danych.

Przedstawione definicje noszą charakter przede wszystkim „techniczny". Funkcjonalną interpretację pojęcia bazy danych podaje J. Martin w cytowanej już pracy [3] na s. 19: „Baza danych może być zdefiniowana jako zestaw powiązanych danych przechowywanych razem **bez szkodliwej lub zbędnej redundancji w celu optymalnej obsługi jednego lub więcej zastosowań;** dane są przechowywane w taki sposób, aby były **niezależne od programów,** które z nich korzystają. Ponadto autor ten zaleca wycofanie z obiegu pojęcia „bank danych", które często ma przeciwstawne interpretacje (jedni specjaliści uważają, że bank danych jest zestawem baz, zaś inni, że baza jest zestawem banków danych), i proponuje używanie wyłącznie terminów „baza danych" i „system baz danych", traktując ten ostatni termin jako zbiorcze określenie kilku baz danych. W tym miejscu pozwalamy sobie wtrącić uwagę, że istnieje potrzeba ujęcia jednym terminem „całości" składającej się z bazy danych i oprogramowania związanego z jej eksploatacją, niezależnie od tego ile baz danych występuje. Do tego celu termin „bank danych" wydaje się być przydatny.

STANDARDOWA ORGANIZACJA ZBIORÓW NA TLE WYMOGÓW BAZY DANYCH

Przez standardową organizację zbiorów rozumiemy tutaj oprogramowanie obsługi zbiorów, dostarczane przez system operacyjny komputera. Jako przykład takiego oprogramowania przyjmijmy system DOS maszyn Jednolitego Sytemu. Omawiane będą dwie podstawowe organizacje dostępu: indeksowo-sekwencyjnego i przypadkowego. Ze

[1]) Czasem funkcje systemu operacyjnego przejmuje system zarządzania bazą danych

względu na brak bezpośredniego dostępu pominięto organizację sekwencyjną, aczkolwiek może być ona wykorzystywana do tworzenia zbiorów operacyjnych systemu zarządzania bazą danych (np. dziennika użytkowań bazy, zbioru zmian, itp.). Jako ilustrację metody pokonywania ograniczeń systemu operacyjnego posłużą nam rozwiązania pakietu IMS firmy IBM.

Za kryteria oceny przydatności standardowych organizacji zbiorów do realizacji celów stawianych przed SZBD przyjmujemy najczęściej:

— możliwość odwzorowania struktur logicznych, w szczególności struktur hierarchicznych i sieciowych
— spełnienie wymogów dopuszczalnego czasu aktualizacji i wymaganego czasu odpowiedzi
— zajętość pamięci dyskowej i operacyjnej
— łączenie zalet przetwarzania partiowego i w czasie rzeczywistym (w trybie pytanie — odpowiedź).

W organizacji dostępu indeksowo-sekwencyjnego kolejność zapisów może być wykorzystana do przedstawienia struktur liniowych i prostych struktur hierarchicznych (patrz artykuł pt. „Przegląd klasycznych i nowoczesnych organizacji zbiorów". INFORMATYKA nr 4/78). Sposób ten nazywa się odwzorowaniem fizycznym. W przypadku złożonych struktur hierarchicznych (np. w schemacie montażowym wyrobów) szersze wykorzystanie organizacji indeksowo-sekwencyjnej wymaga wyjścia poza system operacyjny. Przykładowo, w pakiecie IMS *(Information Management System)* stosowane są własne metody:
a) HSAM — Hierarchical Sequential Access Method
b) HISAM — Hierarchical Indexed Sequential Access Method.

W organizacji dostępu przypadkowego (ang. *random)* przy adresowaniu pośrednim do określenia zależności logicznych nie może być wykorzystana kolejność zapamiętywania zapisów, gdyż jest ona z punktu widzenia logicznych struktur danych zupełnie przypadkowa. Zapisy włączane są do zestawów logicznych poprzez więzi adresowe (łańcuchy) lub skorowidze adresowe. Można więc tworzyć różnorodne struktury, przy czym niecelowe jest odwzorowywanie struktur liniowych, które można efektywniej zrealizować w organizacji indeksowo-sekwencyjnej. System operacyjny nie dostarcza aparatu łańcuchowania logicznego.

W IMS występują dwie modyfikacje organizacji dostępu przypadkowego:
a) HIDAM — Hierarchical Indexed Direct Access Method
b) HDAM — Hierarchical Direct Accect Method.

Jak wynika z powyższych rozważań, standardowe oprogramowanie zbiorów jest z reguły niewystarczające do utrzymywania bazy danych. System operacyjny bowiem nie zabezpiecza powiązań pomiędzy elementami logicznych struktur danych, jak również nie w pełni — w przypadku zapisów segmentowych — zaspokaja potrzeby utrzymywania fizycznych struktur. Ponadto znacznym utrudnieniem w realizacji postulatu niezależności danych jest w systemie DOS „przywiązanie" zbiorów do fizycznych identyfikatorów urządzeń oraz brak możliwości deklarowania formatu zapisu i bloku poprzez karty sterujące zadaniem (Job Control).

KLASYFIKACJA BAZ I BANKÓW DANYCH

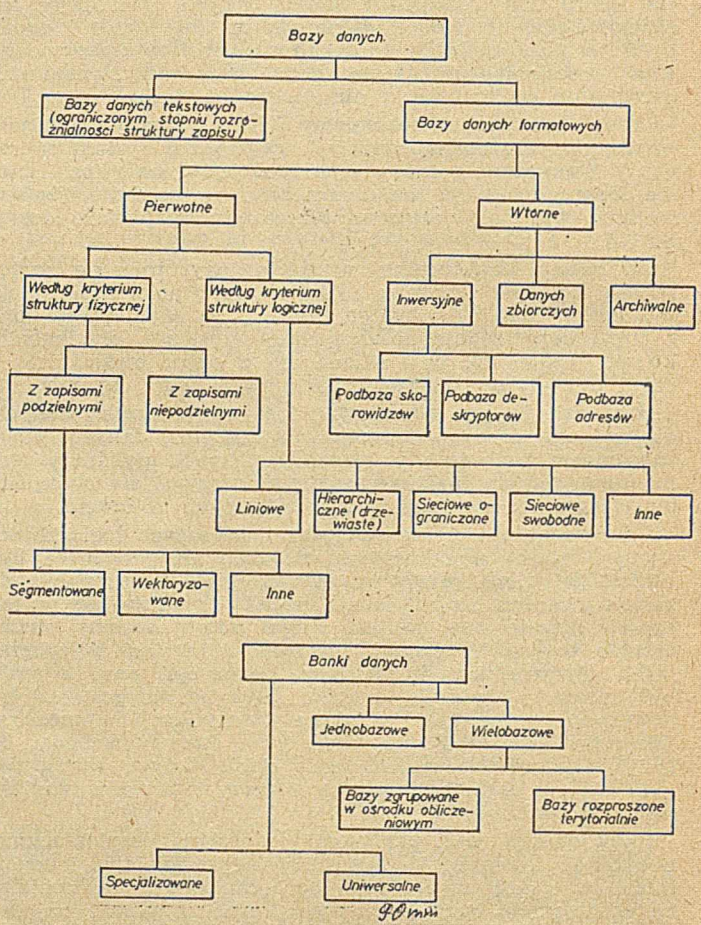
Nie istnieje powszechnie stosowana systematyka baz danych. Na rys. 1 proponujemy ich podział wg cech, które wydają się najbardziej istotne z punktu widzenia projektanta systemu. Z bardziej szczegółowego omówienia wyłączamy bazy danych tekstowych, stosowane np. w wyszukiwaniu informacji bibliograficznych, jako mało przydatne do przetwarzania danych administracyjno-ekonomicznych, charakteryzujących się precyzyjnym opisem strukturalnym.

Klasyfikacja baz danych formatowych rozpoczyna się od podziału na bazy pierwotne i wtórne. **Bazy pierwotne** zasilane są bezpośrednio danymi źródłowymi i stanowią podstawowe źródło informacji analitycznych. **Bazy wtórne** noszą charakter optymalizujący i powstają na podstawie baz pierwotnych. Wśród baz wtórnych wyróżnia się:

a) bazy inwersyjne — optymalizują wyszukiwanie informacji z baz pierwotnych wg kluczy wtórnych (atrybutów)

b) bazy danych zbiorczych — jak wynika z nazwy służą do przechowywania danych zbiorczych wg agregacji czasowych (dekadami, miesiącami, kwartałami, latami, itp.), organizacyjnych (gniazdami, wydziałami, pionami, przedsiębiorstwami, itp.) i rzeczowych (wg pełnego symbolu indeksowego, grup wyrobów lub materiałów, produkcji globalnej w skali przedsiębiorstwa, itp.); bazy te pozwalają na wyeliminowanie wielokrotnego czytania baz pierwotnych, jednakże aktualizacja danych zbiorczych może być złożona i czasochłonna

c) bazy archiwalne — przeznaczone są do gromadzenia wybranych informacji przedterminowych z punktu widzenia bieżącej działalności, celem przejściowego odciążania baz pierwotnych oraz ich zabezpieczenia drogą rekonstrukcji; znaczenie baz archiwalnych rośnie w miarę upływu czasu eksploatacji systemu informatycznego, gdyż gromadzenie przez wieloletni okres wszystkich informacji w bazach pierwotnych staje się często niemożliwe m. in. w wyniku ograniczeń pojemności pamięci dyskowych.



Rys. 1. Klasyfikacja baz i banków danych

Dalszy podział baz danych przebiega wg struktur fizycznych i logicznych, przy czym nie są to podziały rozdzielne lecz łączne (każda baza charakteryzuje się określoną strukturą fizyczną i logiczną). Struktury danych omawialiśmy w artykule pt. „Przegląd klasycznych i nowoczesnych organizacji zbiorów", INFORMATYKA nr 4/78. Zatrzymajmy się jednak na kwestii segmentacji zapisów, jako często spotykanej metodzie projektowania struktur fizycznych w warunkach banku danych (np. w IMS).

Segment fizyczny jest jednostką strukturalną zapisu, która ma zwykle stałą długość i stanowi element transferu danych pomiędzy programem a bazą danych. Format segmentu można ustalać kierując się następującymi kryteriami:

— optymalnym rozmieszczeniem na dysku (uzyskanie współczynnika blokowania zapewniającego najlepsze wykorzystanie ścieżki)
— pojemnością informacyjną dokumentu lub zdarzenia informacyjnego
— grupowaniem danych o zbliżonych powiązaniach logicznych lub podobnej częstotliwości aktualizacji i wyszukiwania
— innymi kryteriami (np. ograniczenia wiersza drukarki lub linii monitora ekranowego w przypadku segmentów wykorzystywanych głównie na wyjściu).

Potrzeba segmentacji zapisu jest oczywista, gdy np. okaże się, że przekracza on pojemność ścieżki dyskowej. Wprowadzenie techniki segmentowania jest szczególnie korzystne w przypadku systemów rozwijanych, w których zachodzi potrzeba sukcesywnej rozbudowy zbiorów o dodatkowe dane.

Rozróżnia się następujące metody segmentacji:
a) segmentację techniczną
b) segmentację logiczną
c) segmentację fizyczną.

Segmentacja techniczna danych występuje w przypadku, gdy zapis znajduje się w kilku blokach. Format taki dopuszczalny jest przez kompilator języka COBOL na maszyny Jednolitego Systemu. Segmentem w tej metodzie jest fragment zapisu znajdujący się w jednym bloku. Program użytkowy ma dostęp do pełnych zapisów.

Metoda segmentacji logicznej znalazła zastosowanie w bazach danych definiowanych w języku PL/1 firmy IBM, wykorzystywanym w pakietach IMS, CICS i MARK IV. Pozwala ona na tworzenie nie tylko prostych segmentów, danych, zwanych segmentami fizycznymi, lecz również segmentów logicznych (w skład których wchodzą dane z różnych segmentów fizycznych) i zapisów logicznych. **Segment fizyczny** jest ciągiem kolejnych pól, zwykle ze sobą powiązanych, np. segment danych stałych wyrobu zawiera symbol wyrobu, nazwę, numer rysunku montażowego, ciężar jednostki, symbol jednostki miary, itp. Segment ma określony identyfikator i co najmniej jedno z jego pól spełnia kryterium porządkowania (przy tworzeniu bazy segmenty zapisu uporządkowane są wg hierarchicznej kolejności zgodnie z zasadą „od najwyższego poziomu do najniższego i od lewej do prawej strony").

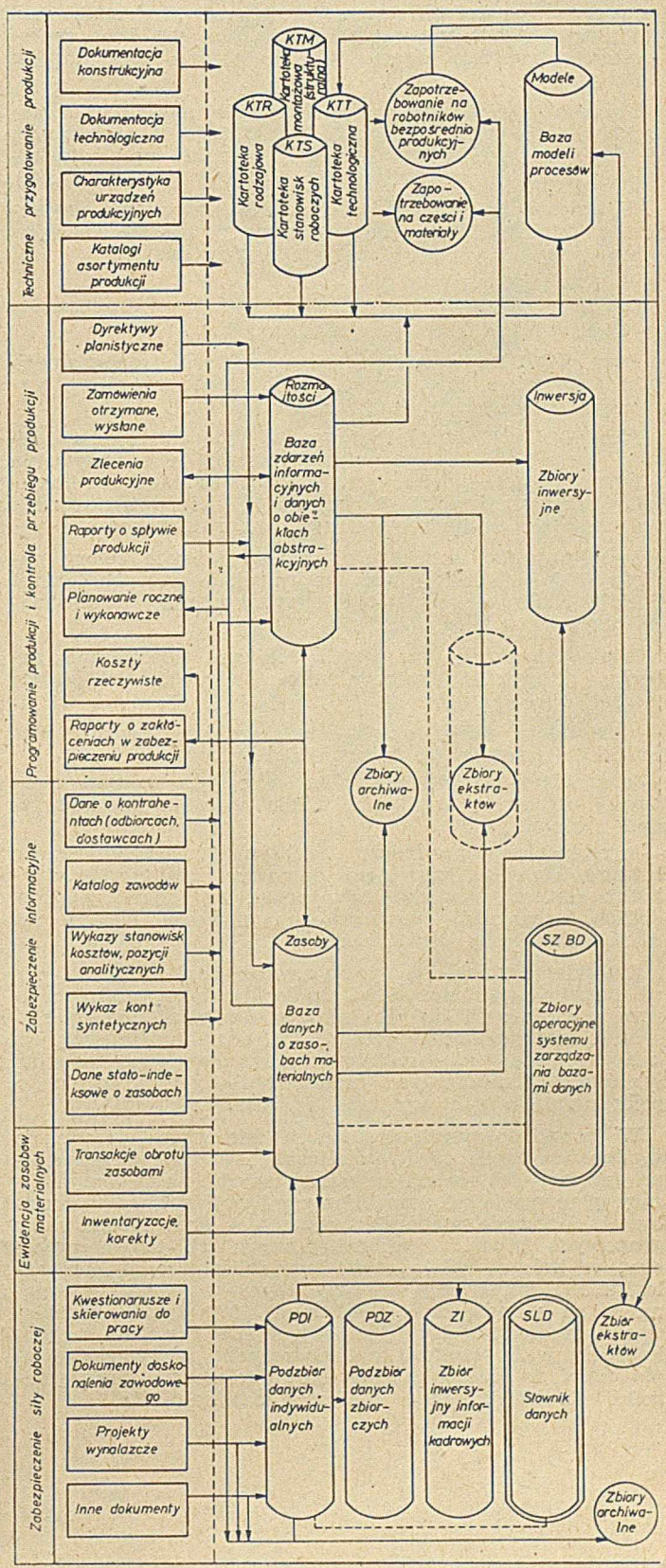
System klasyfikacji banków danych (patrz rys. 1) oparty może być na kryterium stopnia uogólnienia i liczby baz danych. **Banki danych specjalizowane tematycznie** mają zwykle uproszczony system zarządzania bazą, nakładający ograniczenia zarówno na zbiór transakcyjny (dopuszczalne są tylko określone typy zapisów), jak i na bazy danych. **Banki danych uniwersalne** (budowane np. wg koncepcji DBTG-CODASYL) — jak wynika z nazwy — mogą być wykorzystywane w różnych zastosowaniach, aczkolwiek nie zawsze z równie dobrą efektywnością.

Do kluczowych decyzji projektanta systemu wyposażonego w bank danych należy podział zasobów informacyjnych pomiędzy bazy danych, względnie ulokowanie ich w jednej wspólnej bazie danych. Idea wspólnej bazy danych zaprezentowana była w poprzednim artykule naszego cyklu i nie będziemy jej tu poświęcać miejsca.

Wielobazowy bank danych składa się z baz i zbiorów systemowych wykorzystywanych przez system zarządzania bazami lub administratora baz danych. Do zbiorów systemowych należeć mogą m. in. słowniki danych (przechowujące opis danych), zbiory ekstraktów, zbiory archiwalne zabezpieczające. Liczba baz danych zależna jest od czynników technologicznych (w tym ograniczeń pakietu banku danych), wielkości strumienia informacji, układów ewidencyjno-rozliczeniowych i dostępnej konfiguracji komputera. Może ona się zmieniać stosownie do rozbudowy systemu, możliwości technicznych użycia baz rozproszonych (łączonych poprzez sieci teletransmisji), itp.

Wśród czynników technologicznych stanowiących kryterium doboru baz danych wyróżnić można:
— jednorodność struktury fizycznej (np. wszystkie zapisy bazy podlegają segmentacji)
— jednorodność struktury logicznej (np. każdy zapis bazy może wystąpić jako element sieci logicznej)
— liczebność powiązań pomiędzy danymi oraz kierunki powiązań
— technika powiązań (np. w technice inwersyjnej powstaje odrębna wtórna baza danych obsługiwana przez specyficzną technologię).

Wśród układów ewidencyjno-rozliczeniowych stanowiących istotny czynnik podziału na bazy danych w przedsiębiorstwie przemysłowym wymienić można:



Rys. 2. Ramowy schemat wielobranżowego banku danych do zarządzania produkcją

— grupowanie danych wg zasobów i funkcji (procesów), np. zaopatrzenie materiałowe, sprzedaż wyrobów

— grupowanie danych według obiektów nie będących zasobami lecz występujących jako podmioty zdarzeń gospodarczych (obiektami takimi są dostawcy, odbiorcy)

— grupowanie danych według zdarzeń gospodarczych typu zamówienie, zlecenie produkcyjne, dostawa, itp.

— grupowanie danych wg różnorodnych atrybutów charakteryzujących obiekty i zdarzenia (np. wg kont syntetycznych, komórek organizacyjnych, pozycji analitycznych kosztów itp.).

Ponadto istnieją w przedsiębiorstwie układy informacyjne umożliwiające obliczenie wskaźników i parametrów dotyczących przedsiębiorstwa jako całości (np. wskaźnika rentowności, produktywności środków trwałych, itp.).

W proponowanej na rys. 2 metodzie rozwiązania projektowego wielobazowego banku danych zasoby informacyjne gromadzone są w następujących bazach danych:

a) w bazie ZASOBY
b) w bazie ROZMAITOŚCI
c) w bazach pakietów specjalizowanych.

Baza ZASOBY grupuje dane o zasobach materialnych przedsiębiorstwa (materiałach, elementach wyrobów, wyrobach, urządzeniach produkcyjnych, itp.).

Baza ROZMAITOŚCI służy głównie do ujmowania zdarzeń gospodarczych (zamówień, zleceń), przechowywania informacji stało-indeksowych o kontrahentach i obiektach „abstrakcyjnych" (typu: zawód, konto syntetyczne itp.) oraz do gromadzenia danych wskaźnikowych.

Bazy pakietów specjalizowanych to głównie zbiory pakietu BOMP (STEP) i baza informacji kadrowych. Pakiet BOMP (STEP) ukierunkowany jest na obsługę technicznego przygotowania produkcji.

Bank informacji kadrowych realizuje specyficzne zadania gospodarki zatrudnieniowo-kadrowej, polegające m. in. na:

— stosowaniu wielu atrybutów charakteryzujących pracownika (z reguły sto kilkadziesiąt informacji)

— sporządzaniu wielu zestawień zbiorczych (w tym sprawozdawczych) wymagających wielu sortowań w klasycznym systemie

— występowaniu potrzeby selektywnego wyszukiwania informacji według kombinacji dowolnych praktycznie atrybutów (np. wg zawodu, stażu pracy, znajomości języków obcych, uprawnień, odznaczeń, wykształcenia, miejsca zamieszkania, liczby dzieci, itp.).

Omawiając strukturę wielobazowego banku danych, nie od rzeczy będzie zaznaczyć, że w systemie informatycznym poza bazami danych istnieją oczywiście zbiory klasyczne zorganizowane, przeznaczone dla mniej znaczących danych lub powstałe na okres przejściowy (np. kartoteki przedmiotów nietrwałych przeznaczenia nieprodukcyjnego, zbiór przejściowy z wynikami inwentaryzacji, itp.).

W zaawansowanych systemach kierowania produkcją występować może specyficzna **baza MODELE**, przeznaczona głównie do obsługi matematycznych metod optymalizacji planowania produkcji (optymalizacji zużycia materiałów i obciążenia stanowisk roboczych) oraz modelowania grupowych procesów technologicznych, tj. takich procesów, w których wiele części ma zbliżone cechy technologiczne i konstrukcyjne, przy czym mogą być wytwarzane na urządzeniach zamiennych. W wyniku modelowania procesu możliwe jest otrzymanie kartoteki technologicznej jako rezultatu automatycznego przyporządkowania części do maszyn wg kryterium tolerancji, klasy gładkości, obróbki cieplnej, wielkości materiału, itp.

Eksploatacja bazy MODELE, będącej w zasadzie zbiorem algorytmów liniowego i dynamicznego programowania oraz macierzy współczynników, powinna przebiegać w warunkach wielobazowego banku danych znacznie sprawniej niż w przypadku obliczeń optymalizacyjnych w klasycznym systemie. Wynika to z możliwości zasilania modeli optymalizacyjnych przez wszystkie (w miarę potrzeby) bazy bez konieczności każdorazowego przygotowywania danych wejściowych do obliczeń.

Podstawowe bazy banku danych powinny być uzupełniane **zbiorami inwersyjnymi**, przeznaczonymi z reguły do sygnalizowania informacji odchyleniowych i zabezpieczenia szybkiego dostępu do pierwotnej bazy wg kluczy wtórnych.

W przypadku podjęcia decyzji budowy wielobazowego banku danych do rozstrzygnięcia pozostaje problem lokalizacji tych baz. Bazy zlokalizowane (scentralizowane) w ośrodku obliczeniowym są typowym rozwiązaniem dla przedsiębiorstw. W systemach funkcjonujących w skali zjednoczeń lub resortów pod uwagę należy wziąć również **zagadnienie rozproszonych terytorialnie baz danych.** Przetwarzanie w systemie opartym na rozproszonych bazach danych przebiega na nieco innych zasadach, gdyż poza technicznym problemem zdalnej komunikacji pomiędzy bazami występują kwestie selekcji wybranych informacji analitycznych (szczególnie odchyleniowych) oraz agregacji danych stosownie do potrzeb zarządzania na wyższych szczeblach organizacyjnych. Bez ich uwzględnienia koszty transmisji danych pomiędzy bazami z uwagi na wielkość strumienia danych będą bardzo duże, zaś przydatność informacji niewielka.

* * *

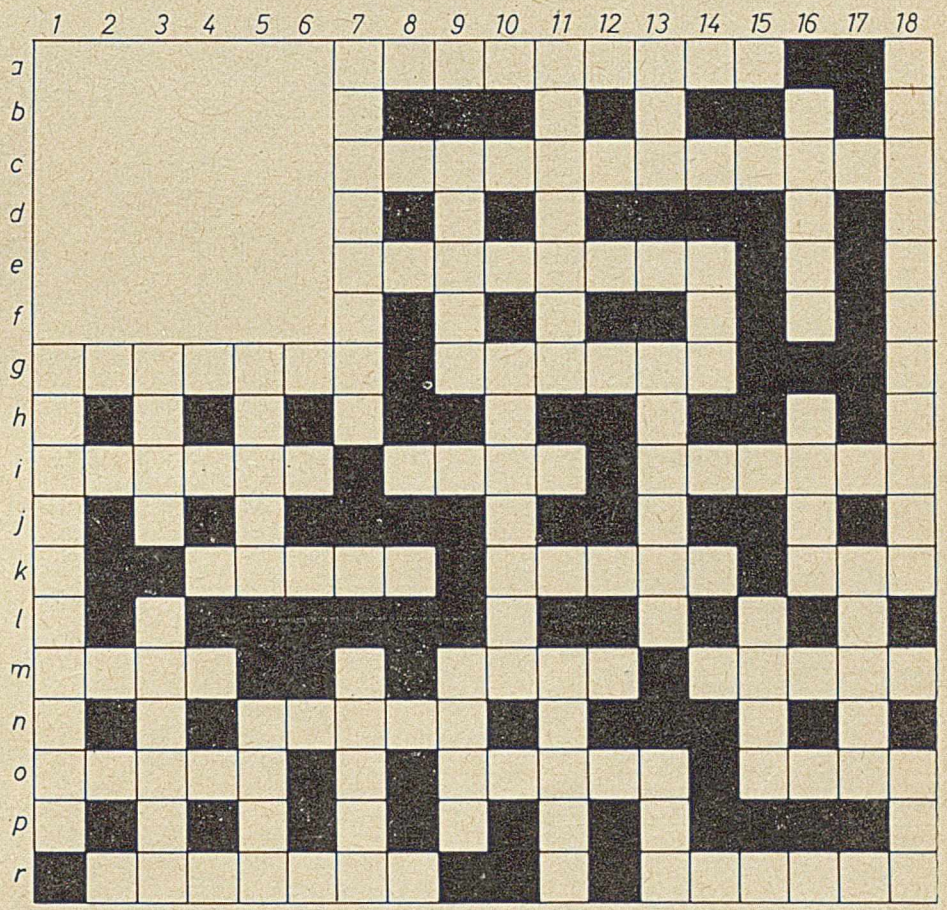
Bazy danych są nowym jakościowym krokiem w usprawnieniu organizacji zasobów informacyjnych skomputeryzowanego systemu. Umożliwia on pokonanie bariery „sztywnych" zbiorów danych i pozwala na wprowadzenie elastycznych logicznych struktur danych odwzorowujących zjawiska gospodarcze zachodzące w przedsiębiorstwie.

W niniejszym artykule przedstawiliśmy między innymi koncepcję wielobazowego banku danych. Jest to oczywiście jedna z wielu możliwych propozycji przejścia na wprowadzenie logicznych struktur danych i systemu zarządzania bazą danych jako podstawy realizacji nowoczesnych zastosowań. W kolejnych artykułach cyklu postaramy się szczegółowiej omówić systemy zarządzania bazą danych i zaproponować również inne koncepcje banków danych.

LITERATURA:

[1] CODASYL: Język opisu danych (tłum. z ang.). OBRI, Warszawa 1977
[2] CODASYL: Raport grupy roboczej do spraw baz danych (tłum. z ang.). OBRI, Warszawa 1977
[3] J. Martin: Computer data-base organization. Prentice-Hall, 1975
[3] H. Wedekind: On the parametric specification of data-base oriented information systems. Management Datamatics, 5/1976
[5] IBM: Information Management System. IMS/360. General Information Manual GH-20-0765-4
[6] W. E. Bender: Introductory manual to data base management. Management Informatics, vol. 3 (1974), no 2
[7] Z. Ryznar: Projektowanie wspólnej bazy danych dla potrzeb przemysłu. INFORMATYKA 1/74
[8] Diebold (Europejski Program Badawczy): Organizing for data base management. Doc. E99 1972 (tłum. OBRI zeszyt 49. 1974)
[9] Diebold (Europejski Program Badawczy): IMIS implementation. European experience. Doc. E78 (tłum. OBRI wdrażanie zintegrowanych systemów informowania kierownictwa. Doświadczenia europejskie. Zeszyt 41. 1973)

Krzyżówka 015



Poziomo:

a7 — transmisja danych
c7 — operacja odtworzenia pierwotnej postaci danych z istniejącej postaci danych uprzednio upakowanych
e7 — firma komputerowa produkująca między innymi minikomputery oraz mikroprocesory pracujące w logice identycznej jak PDP-8
g1 — sytuacja, w której następuje działanie uniemożliwiające dostęp do zasobów systemu
g9 — działo
i1 — dawna nazwa najemnego robotnika rolnego
i8 — popularna maszyna analogowa produkcji czechosłowackiej
i13 — stan w środkowej Indii
k4 — standardowa funkcja zewnętrzna, obliczająca pierwiastek kwadratowy z liczby zespolonej w języku FORTRAN
k10 — znany producent mikroprocesorów oraz systemów wieloprocesorowych
k16 — np. na Centrum Zdrowia Dziecka
m1 — europejska agencja prasowa
m9 — umowna liczba bitów, przyjęta w organizacji niektórych komputerów jako pojemność podstawowej komórki pamięci
m14 — nieprzekraczalna granica
n5 — dawna nazwa Malawi
o1 — jeden z podstawowych nośników informacji w technice komputerowej
o9 — twórca teorii zbiorów rozmytych
o15 — nazwa pierwszego w świecie systemu wielodostępnego
r2 — sieć komputerowa współpracująca z komputerem COMNET-2
r13 — znany chemik-organik, matematyk, profesor Politechniki Warszawskiej (1879—1944)

Pionowo:

1g — zbiór sprawdzonych programów dla określonego typu komputera
3g — kwant czasu pracy procesora przydzielanego poszczególnym procesom, podczas którego proces jest aktywny
3l — koncepcja maszyny niejednorodnej, opartej na uniprocesorze IBM-7094
5g — pochodząca z Afryki roślina uprawiana dla swych właściwości leczniczych i ozdobnych
5n — pakt północnoatlantycki
7a — urządzenie końcowe maszyny cyfrowej
7m — amerykański wieloprocesorowy system rezerwacji miejsc lotniczych
9c — producent transformatorów podległy zjednoczeniu UNITRA
9m — grupa zbiorów danych
10g — grupa komputerów, w której obowiązuje jednolitość języka wewnętrznego
11a — rysunek przedstawiający zależności między wielkościami
11m — część programów sterujących, rezydująca stale w PAO podczas pracy systemu
13a — jeden ze sposobów ochrony wewnętrznego rynku towarowego
13g — sieć komputerowa, w której komputerem komunikacyjnym jest minikomputer VARIAN DATA-620
13o — czeski reformator religijny
14e — obszar PAO przeznaczony na moduły wielodostępne (obszar łączenia lub obszar modułów rezydentnych)
15l — masowe zgromadzenie ludności
16b — dyrektywa języka FORTRAN, umożliwiająca dostęp do dowolnego punktu segmentu programu
16h — rodzaj opadu atmosferycznego
17k — nazwa zastępcza członu zbioru bibliotecznego
18a — program realizujący wstępne funkcje obliczeniowe lub organizacyjne
18o — jeden z języków służących do symulacji procesów.

Hasło:

g6 o9 m15 h18 b13 c8 e14 c17 m18 h18 e16 a15 a13 c 12 e13 j5 c14 d11 e18 p13 r6 k4 k1 o12 c7 c9 e10 o15 o9 h13 k11 m15 o12 a14 a10 r13 o16 r6 k1 o15 r14 h13 i8 m15 r6 r3 f13 g11 a14 c15 o1 p9 k14 r4 k17 m9 k13 d16 c18 i13 r17 p18 r14 k7 c9 p13 k4 i5 r6 f13 m16 c14 n5 h1 b7 b13 a15 i4 o9 m15 h18. a14 a10 n17 e8 k4 i5 o15 l13 c18 a15 p13

Rozwiązania prosimy nadsyłać do dnia 30 października br. pod adresem: redakcja „Informatyki", ul. Jasna 14/16, 00-041 Warszawa. Wśród Czytelników, którzy nadeślą prawidłowe odpowiedzi, zostaną rozlosowane nagrody książkowe.